



ОГЛАВЛЕНИЕ

| | |
|-------------------------------------------------------------|-----------|
| Предисловие | 16 |
| Вступление | 19 |
| Чему я научусь? | 20 |
| В чем состоят особенности iPad? | 21 |
| Большой экран | 21 |
| Менее глубокие иерархии | 22 |
| Ориентация имеет значение | 25 |
| Мультисенсорность наступает | 26 |
| iPad ликвидирует разрыв между телефоном и компьютером | 27 |
| Здравствуй, iPad | 28 |
| Структура книги | 28 |
| На кого рассчитана эта книга | 34 |
| Получение исходного кода PhotoWheel | 34 |
| Благодарности | 36 |
| Благодарности Кирби Тэрнера | 36 |
| Благодарности Тома Харрингтона | 36 |
| Об авторах | 38 |
| ЧАСТЬ I. ПРИСТУПАЯ К РАБОТЕ | 39 |
| Глава 1. Первое приложение | 40 |
| Создание проекта «Здравствуй, мир» | 40 |
| Вывод текста на экран | 48 |
| Поздоровайся с дядей | 50 |
| Резюме | 55 |
| Глава 2. Знакомство с Xcode | 56 |
| Интегрированная среда разработки | 57 |
| Рабочее пространство | 57 |
| Панель инструментов | 58 |
| Область навигации | 59 |
| Область редактора | 60 |
| Область вспомогательных средств | 61 |
| Область отладки | 63 |
| Настройки | 64 |
| Шрифты и цвета | 64 |
| Редактирование текста | 65 |
| Настройки клавиш | 68 |
| Автозавершение кода | 70 |
| Документация для разработчика | 71 |

| | |
|------------------------------------------------------|------------|
| Редакторы | 72 |
| Параметры проекта | 74 |
| Схемы | 76 |
| Организатор | 77 |
| Другие инструменты Xcode | 78 |
| Резюме | 79 |
| Глава 3. Знакомство с Interface Builder | 81 |
| Interface Builder | 81 |
| Как работает IB? | 82 |
| Практическое занятие | 83 |
| Выбор и копирование объектов | 87 |
| Выравнивание объектов | 87 |
| Компоновочный прямоугольник | 90 |
| Изменение состояния | 91 |
| Связывание NIB-файла с кодом | 94 |
| Определение выхода в коде | 95 |
| Использование разделенного редактора Assistant | 99 |
| Раскадровки | 101 |
| Резюме | 102 |
| Глава 4. Знакомство с Objective-C | 103 |
| Что такое Objective-C? | 103 |
| Практикум по Objective-C | 104 |
| Пишем код | 107 |
| Объект | 109 |
| Класс | 109 |
| NSObject | 112 |
| Интерфейс | 112 |
| Переменные экземпляра | 113 |
| Объявленные свойства | 114 |
| Методы | 117 |
| Реализация | 118 |
| Директива synthesize | 119 |
| init | 120 |
| super | 121 |
| flip | 121 |
| Селекторы | 122 |
| Синтаксис с точками | 124 |
| Использование класса CoinTosser | 124 |
| Управление памятью | 126 |
| Автоматический подсчет ссылок | 127 |
| Резюме | 128 |
| Глава 5. Знакомство с Сосоа | 129 |
| Стек Сосоа | 129 |
| Подсистема Foundation | 131 |
| Типы данных | 132 |
| Классы коллекций | 138 |

| | |
|-----------------------------------------------------------------------------------------------------|------------|
| Служебные классы и функции | 140 |
| Другие анализаторы XML | 143 |
| Подсистема UIKit..... | 144 |
| Паттерны проектирования в Socoa | 153 |
| Модель-Представление-Контроллер | 154 |
| Цель-действие | 154 |
| Резюме | 155 |
| Глава 6. Подготовка iPad | 156 |
| О портале подготовки для iOS | 156 |
| Процедура подготовки: краткий обзор | 158 |
| Что такое идентификатор устройства? | 159 |
| Что такое идентификатор приложения? | 159 |
| Что такое профиль подготовки к разработке?..... | 161 |
| Настройка компьютера, используемого для разработки..... | 162 |
| Запрос сертификата разработки..... | 163 |
| Отправка CSR на рассмотрение | 166 |
| Скачивание и установка сертификата..... | 168 |
| Настройка устройства | 170 |
| Использование устройства для разработки..... | 170 |
| Использование портала подготовки для iOS | 173 |
| Добавление идентификатора устройства | 173 |
| Добавление идентификатора приложения | 175 |
| Создание профиля подготовки к разработке | 177 |
| Скачивание профиля подготовки к разработке..... | 179 |
| Установка профиля подготовки к разработке | 179 |
| Резюме | 181 |
| Глава 7. Проектирование приложений | 182 |
| Определение приложения | 182 |
| Название приложения | 183 |
| Краткое описание приложения..... | 184 |
| Список функций | 184 |
| Целевая аудитория | 186 |
| Конкурирующие продукты | 187 |
| Пример хартии приложения | 188 |
| О проектировании пользовательского интерфейса | 190 |
| Прочитайте рекомендации по разработке человеко-машинного интерфейса..... | 190 |
| Учитывайте наличие сенсорного интерфейса..... | 190 |
| Проектируйте интерфейс для конкретного устройства..... | 191 |
| Пользователи работают с устройствами на базе iOS не так, как с настольными ПК или веб-сайтами | 191 |
| Преобразитесь в промышленного дизайнера..... | 192 |
| Метафоры..... | 193 |
| Звуковые эффекты..... | 194 |
| Настройка имеющихся элементов управления | 195 |
| Наймите дизайнера..... | 196 |
| Имитации | 197 |

| | |
|------------------------------|-----|
| Что такое имитация? | 197 |
| Что имитировать | 199 |
| Инструментарий | 200 |
| Создание прототипа | 205 |
| Что такое прототип? | 206 |
| Как создается прототип | 207 |
| Резюме | 208 |

ЧАСТЬ II. РАЗРАБОТКА ПРИЛОЖЕНИЯ

PhotoWheel..... 209

Глава 8. Создание приложения с интерфейсом

«основной-подробности» 210

| | |
|---------------------------------------------------------------------|-----|
| Создание прототипа | 210 |
| Что такое контроллер разделенного представления? | 212 |
| Создание проекта | 212 |
| Работа с эмулятором | 215 |
| Познакомимся поближе | 216 |
| Структура проекта | 217 |
| Делегат приложения | 218 |
| Параметры запуска | 223 |
| Другие методы в классе UIApplicationDelegate | 225 |
| Знакомство с классом UISplitViewController | 226 |
| Инициализация делегата контроллера разделенного представления | 229 |
| Контроллер подробного представления | 230 |
| Контроллер основного представления | 230 |
| Резюме | 232 |
| Упражнения | 232 |

Глава 9. Табличные представления 233

| | |
|-----------------------------------------|-----|
| Начнем с начала | 233 |
| Присмотримся поближе | 237 |
| Класс UITableView | 238 |
| Класс UITableViewCell | 239 |
| Класс UITableViewDelegate | 239 |
| Класс UITableViewDataSource | 239 |
| Класс UITableViewController | 239 |
| Работа с табличным представлением | 240 |
| Простая модель | 240 |
| Отображение данных | 242 |
| Добавление данных | 249 |
| Редактирование данных | 266 |
| Удаление данных | 271 |
| Переупорядочение данных | 273 |
| Выбор данных | 274 |
| Резюме | 277 |
| Упражнения | 277 |

| | |
|----------------------------------------------------------------|------------|
| Глава 10. Представления | 278 |
| Пользовательские представления | 278 |
| Контроллер или представление? | 279 |
| Круговое представление | 280 |
| Карусельное представление | 288 |
| Ячейка для кругового представления фотографий | 295 |
| Использование PhotoWheelViewCell | 298 |
| Резюме | 300 |
| Упражнения | 301 |
| Глава 11. Жесты | 302 |
| Что такое жесты? | 302 |
| Предопределенные жесты касания | 303 |
| Типы жестов | 304 |
| Как пользоваться распознавателями жестов | 304 |
| Нестандартные жесты касания | 308 |
| Создание распознавателя жеста раскручивания | 309 |
| Использование распознавателя жеста раскручивания | 312 |
| Резюме | 317 |
| Упражнения | 317 |
| Глава 12. Добавление фотографий | 318 |
| Два подхода | 318 |
| Подсистема Assets Library | 318 |
| Контроллер выбора изображения | 320 |
| Работа с контроллером выбора изображения | 321 |
| Списки действий | 324 |
| Использование UIImagePickerController | 328 |
| Сохранение в каталоге отснятых фотографий | 333 |
| Резюме | 334 |
| Упражнения | 334 |
| Глава 13. Сохранение данных | 335 |
| Модель данных | 335 |
| Фотографии | 335 |
| Фотоальбомы | 336 |
| Предусмотрительность полезна | 336 |
| Построение модели на основе списков свойств | 336 |
| Что такое список свойств? | 337 |
| Подготовка модели данных | 337 |
| Чтение и сохранение фотоальбомов | 339 |
| Добавление в альбом новых фотографий | 344 |
| Отображение фотографий в альбоме | 349 |
| Построение модели на основе Core Data | 351 |
| Что такое Core Data? | 351 |
| Управляемые объекты и описания сущностей | 352 |
| Контекст управляемого объекта | 354 |
| Постоянное хранилище и координатор постоянного хранилища | 355 |

| | |
|--------------------------------------------------------------------|------------|
| Добавление Core Data в PhotoWheelPrototype | 355 |
| Добавление средств Core Data | 355 |
| Конфигурирование стека Core Data | 357 |
| Использование Core Data в приложении PhotoWheel | 361 |
| Редактор модели Core Data | 361 |
| Добавление сущностей | 362 |
| Создание подклассов NSManagedObject | 365 |
| Добавление собственного кода в модельные объекты | 370 |
| Чтение и сохранение фотоальбомов с помощью Core Data | 375 |
| Добавление в альбом новых фотографий с помощью Core Data | 378 |
| Отображений фотографий в альбоме с помощью Core Data | 381 |
| Работа с SQLite напрямую | 382 |
| Резюме | 383 |
| Упражнения | 383 |
| Глава 14. Раскадровки в Xcode | 384 |
| Что такое раскадровка? | 384 |
| Использование раскадровки | 385 |
| Сцены | 386 |
| Переходы | 387 |
| Включение раскадровки в PhotoWheel | 388 |
| Рабочее пространство | 388 |
| Добавление главной раскадровки | 391 |
| Задание UIMainStoryboardFile | 393 |
| Внесение изменений в файл AppDelegate | 394 |
| Добавление изображений | 395 |
| Значок приложения | 395 |
| Начальный контроллер представления | 396 |
| Еще одна сцена | 399 |
| Создание перехода | 402 |
| Резюме | 404 |
| Упражнения | 404 |
| Глава 15. Контроллеры представлений – добавим код | 405 |
| Реализация контроллера представления | 405 |
| Переход | 409 |
| Создание нестандартного перехода | 410 |
| Готовим сцену | 410 |
| Реализация нестандартного перехода | 413 |
| Перед компиляцией | 417 |
| Настройка перехода с выталкиванием | 419 |
| Контейнерный контроллер представления | 422 |
| Создание контейнерного контроллера представления | 424 |
| Добавление дочерних сцен | 424 |
| Добавление дочерних контроллеров представлений | 427 |
| Исправление перехода с заталкиванием | 431 |
| Резюме | 432 |

| | |
|---------------------------------------------------------|------------|
| Упражнения | 432 |
| Глава 16. Конструирование главного экрана..... | 433 |
| Повторное использование кода прототипа | 434 |
| Копирование файлов | 435 |
| Модель Core Data | 436 |
| Модификация класса WheelView | 441 |
| Отображение фотоальбомов | 454 |
| Реализация контроллера представления фотоальбомов | 456 |
| Настройка контекста управляемого объекта | 463 |
| Добавление фотоальбомов | 465 |
| Управление фотоальбомами | 466 |
| Выбор фотоальбома | 466 |
| Задание названия фотоальбома | 471 |
| Исправление внешнего вида панели инструментов | 478 |
| Удаление фотоальбома | 479 |
| Улучшаем миниатюру фотоальбома..... | 482 |
| Добавление фотографий | 486 |
| Отображение фотографий..... | 492 |
| Использование класса GridView..... | 503 |
| Создание класса ячейки сеточного представления | 508 |
| Резюме | 512 |
| Упражнения | 512 |
| Глава 17. Создание обозревателя фотографий | 513 |
| Использование прокручиваемого представления..... | 513 |
| Пользовательский интерфейс обозревателя фотографий..... | 523 |
| Запуск обозревателя фотографий..... | 523 |
| Улучшение заталкивания и выталкивания..... | 526 |
| Добавление эффектов, связанных с обрамлением | 534 |
| Масштабирование | 539 |
| Удаление фотографии | 547 |
| Резюме | 555 |
| Упражнение..... | 555 |
| Глава 18. Поддержка поворота устройства..... | 556 |
| Как поддерживаются повороты | 556 |
| Поддерживаемые ориентации | 557 |
| Авторазмер | 558 |
| Специальная обработка поворотов | 559 |
| Поворот сцены с фотоальбомами | 564 |
| Поворот сцены с фотоальбомом | 565 |
| Шлифовка класса WheelView | 567 |
| Поворот представления About..... | 568 |
| Поворот обозревателя фотографий | 568 |
| Устранение оставшихся проблем | 569 |
| Исправление обозревателя фотографий..... | 569 |
| Исправление главного экрана | 575 |
| Стартовые изображения..... | 578 |

| | |
|--------------------------------------------------------|------------|
| Резюме | 581 |
| Упражнения | 581 |
| Глава 19. Печать с помощью AirPrint | 582 |
| Как работает механизм печати | 582 |
| Центр печати..... | 583 |
| Требования к печати..... | 583 |
| API печати | 583 |
| Эмулятор принтера | 587 |
| Резюме | 588 |
| Упражнения | 589 |
| Глава 20. Отправка электронной почты..... | 590 |
| Как это работает..... | 590 |
| Класс MFMailComposeViewController | 592 |
| Класс SendEmailController | 593 |
| Обзор класса SendEmailController | 594 |
| Использование SendEmailController | 597 |
| Резюме | 603 |
| Упражнения | 603 |
| Глава 21. Веб-службы..... | 604 |
| Основы | 604 |
| REST-службы в Socoa..... | 606 |
| Flickr | 607 |
| Добавление работы с Flickr в PhotoWheel..... | 609 |
| Изменение сцены контроллера представления Flickr | 610 |
| Отображение сцены Flickr | 612 |
| Обертывание Flickr API | 615 |
| Асинхронная загрузка фотографий | 622 |
| Реализация класса FlickrViewController..... | 629 |
| И еще одно | 639 |
| Чего не хватает? | 641 |
| Резюме | 641 |
| Упражнения | 642 |
| Глава 22. Синхронизация с iCloud | 643 |
| Синхронизация – это просто | 643 |
| Основные понятия iCloud..... | 644 |
| Файловые координаторы и презентаторы | 645 |
| Классы UIDocument и UIManagedDocument | 645 |
| Глобально доступные постоянные хранилища | 646 |
| И снова о подготовке устройства..... | 647 |
| Конфигурирование идентификатора приложения | 647 |
| Подготовка для работы с iCloud..... | 649 |
| Настройка прав для работы с iCloud | 650 |
| Адаптация PhotoWheel к работе с iCloud | 653 |
| Не синхронизируйте больше, чем необходимо | 653 |
| Транзитные атрибуты в Core Data | 654 |

| | |
|--------------------------------------------------------------------------------------------------|-----|
| Модификация PhotoWheel для работы с iCloud..... | 655 |
| Синхронизация фотографий с iCloud..... | 660 |
| Модификация координатора постоянного хранилища для работы с глобально доступными данными..... | 661 |
| Получение изменений из iCloud | 666 |
| Резюме | 671 |
| Упражнения..... | 671 |
| Глава 23. Создание слайд-шоу с помощью AirPlay... 672 | |
| Варианты подключения внешнего дисплея | 672 |
| Требования к внешним дисплеям со стороны приложения | 673 |
| API для работы с внешним дисплеем | 674 |
| Добавление в PhotoWheel функции слайд-шоу..... | 675 |
| Изменение раскладки | 676 |
| Добавление показа слайд-шоу..... | 677 |
| Управление внешними дисплеями..... | 681 |
| Переход к следующей фотографии | 685 |
| Добавление элементов управления слайд-шоу | 687 |
| Изменение обозревателя фотографий | 690 |
| Замечание о тестировании и отладке | 691 |
| Добавление поддержки AirPlay | 692 |
| Использование AirPlay | 694 |
| Резюме | 695 |
| Упражнения..... | 696 |
| Глава 24. Визуальные эффекты с помощью Core Image..... 697 | |
| Основные понятия Core Image | 697 |
| Введение в CIFilter | 699 |
| Типы фильтров | 700 |
| Использование CIFilter | 701 |
| Анализ изображений | 702 |
| Автоматическое улучшение качества изображения | 703 |
| Распознавание лиц | 704 |
| Добавление эффектов Core Image в PhotoWheel | 705 |
| Новые методы делегатов | 706 |
| Переменные экземпляра для управления фильтрами..... | 707 |
| Модификация пользовательского интерфейса..... | 708 |
| Создание эффектов CIFilter | 715 |
| Применение фильтров | 720 |
| Реализация автоматического улучшения качества | 721 |
| Реализация панорамирования лиц..... | 721 |
| Прочие необходимые методы | 723 |
| Резюме | 725 |
| Упражнения..... | 725 |
| ЧАСТЬ III. ЗАВЕРШАЮЩИЕ ШТРИХИ 727 | |
| Глава 25. Отладка 728 | |

| | |
|------------------------------------------------------------------------------------------------------|------------|
| Понять, в чем проблема..... | 728 |
| Что не так? | 728 |
| Воспроизведение ошибки | 729 |
| Основные понятия отладки..... | 729 |
| Точки прерывания | 730 |
| Отладка в Xcode..... | 731 |
| Установка и управление точками прерывания | 731 |
| Настройка точек прерывания | 732 |
| Останов в точке прерывания | 734 |
| Просмотр значений переменных..... | 736 |
| Пример отладки: код вывода на внешний дисплей..... | 739 |
| Когда без NSLog не обойтись | 743 |
| Профилирование кода с помощью инструментальных средств | 746 |
| Пример профилирования: обновление элементов управления в пользовательском интерфейсе слайд-шоу | 749 |
| Резюме | 753 |
| Глава 26. Распространение приложения..... | 754 |
| Способы распространения..... | 754 |
| Сборка для разового распространения | 755 |
| Подготовка к разовому распространению | 755 |
| Подготовка сборки для разового распространения..... | 756 |
| Сборка для распространения через App Store | 759 |
| Подготовка для App Store | 759 |
| Подготовка сборки для распространения через App Store..... | 760 |
| Последующие шаги | 762 |
| Процедура размещения в App Store..... | 763 |
| Что делать, если Apple отклоняет заявку?..... | 764 |
| Информация о приложении для App Store | 765 |
| Ресурсы App Store | 768 |
| Использование iTunes Connect | 769 |
| Роли пользователей..... | 769 |
| Управление приложениями | 770 |
| Подача заявки | 770 |
| Что дальше? | 773 |
| Резюме | 774 |
| Глава 27. Послесловие | 775 |
| Что дальше? | 776 |
| Приложение А. Установка средств разработки | 777 |
| У участников программы есть привилегии | 777 |
| Присоединение к программе iOS Developer Program | 778 |
| Какую программу выбрать? | 779 |
| Что нужно для регистрации | 781 |
| Скачивание Xcode | 783 |
| Установка Xcode | 783 |
| Предметный указатель..... | 785 |



ПРЕДИСЛОВИЕ

Книги – это прекрасно, вы согласны?

Все мои знакомые в курсе, что я настоящий книжный червь. Я обожаю книги. Хорошо написанная книга – один из самых дешевых и быстрых способов самообразования. Могу назвать целый ряд книг, оказавших огромное влияние на мое личностное и профессиональное развитие, например: «Object Oriented Software Construction» Бертрана Мейера (Bertrand Meyer), ранние книги по программированию Мас'ов Скотта Настера (Scott Knaster) и Стивена Черникова (Stephen Chernicoff), книги по программированию на С Дейва Марка (Dave Mark), труд Роберта К. Мартина с ужасающим названием (но полный пассажей, прочитав которые, так и хочется воскликнуть «вот оно!») «Designing Object Oriented C++ Applications Using the Booch Method» и, конечно, книги покойного У. Ричарда Стивенса по сетевому программированию и программированию в UNIX. До сих пор помню, какие уроки я вынес из этих объемистых томов, хотя читал их свыше 20 лет назад.

К сожалению, не все книги так хороши. Мне встречалось и никуда не годное барахло. Когда я переходил от программирования Мас'ов к программированию iPhone, мне попадались отличные книги. И безобразные тоже. Просто ужасные – как будто кто-то взял «Краткое руководство по программированию на Visual Basic для полных чайников за 24 часа» и вклеил в него картинки, относящиеся к iPhone. Как-то в книжном магазине я перелистывал одну из первых книг по iPad, в которой ляпы были буквально на каждой странице. Иногда просто опечатки. А иногда тонкие ошибки, понятные лишь тому, кто прожил пару лет в мире Сосоа. Встречались откровенно плохие советы, дать которые мог лишь человек, вообще не понимающий, о чем идет речь. Покупатель, отдающий заработанные тяжким трудом деньги за книгу, рассчитывает получить что-то взамен, и обманывать его чаяния непростительно.

Но вот вы держите в руках книгу «Изучаем программирование для iPad». Стоит ли она своих денег? В какую категорию она попадает: классные или отстойные? Хороший вопрос.

Во-первых, хорошая книга должна раскрывать заявленную тему – и раскрывать полно. Судя по весу, материала в книге много. Ну это, конечно, если вы держите в руках печатное издание. В электронной-то форме «Война и мир» весит столько же, сколько «Маленький принц». Но пролистайте оглавление, и вы убедитесь, что книжка насыщенная. И всё по делу: установка средств разработки, паттерн модель-представление-контроллер, элемент с интерфейсом «основной-подробности», табличные представления, `UITableViewController`, навигационные представления, обработка поворачивания устройства. А также более сложные темы: потребление веб-служб, мультимедийная библиотека, жесты сенсорного интерфейса, сохранение данных. И такие малоприятные вещи, как подготовка устройств Apple перед началом разработки. Освещены и самые последние новинки: раскадровки, технологии AirPrint, AirPlay, iCloud и Core Image. Кирби и Том намутились, пока работали с предварительными версиями программного обеспечения, зато избавили вас от этих страданий.

Очень хорошие книги выходят вовремя, но непригодны для работы. Первую книгу о программировании iPad я увидел спустя три месяца после анонсирования устройства. Но получить из нее целостное представление об iPad читатель не мог, потому что тогда еще никто не держал его в руках. Задача состояла в том, чтобы как можно скорее заявить о себе на рынке, и эта задача была решена. Что же касается этой книги, то сейчас, когда я пишу эти строки, прошло уже больше года с момента начала работы над ней. Чтобы хорошая книга по-настоящему созрела, нужно время.

Выдающиеся книги выходят за рамки заявленной темы. Эта книга называется «Изучаем программирование для iPad». Можно было бы предположить, что это всего лишь примитивное введение на уровне восклицаний «Представления – это круто!», «Ух ты, кнопки-то можно коснуться!». Но этим дело отнюдь не ограничивается. Не так уж много найдется книг, в которых разрабатывается один сквозной проект. Просто потому, что это трудно сделать. Важные библиотечные средства оказываются отнесены куда-то на периферию, так как автор не продумал заранее, что хочет сделать. Но эта книга устроена иначе – сначала проект, потом демонстрационный прототип и, наконец, Вещь.

Но и это не всё. Редко в книге говорится о внутренней кухне проектирования. Здесь говорится. Еще реже авторы снисходят до рассказа об отладке своего детища. Отладка – важнейшая составная часть повседневной работы программиста, а ей обычно уделяют один-два

абзаца. В этой книге отладке посвящена целая глава, отнюдь не ограничивающаяся описанием пошагового выполнения. Читая предварительные варианты рукописи (чтобы написать сколько-нибудь вразумительное предисловие), я аж присвистнул, дойдя до главы 25. Я люблю отлаживаться и мне очень нравится, что эта важная тема так подробно разобрана в книге, на первый взгляд адресованной начинающим. А еще я люблю учиться. И из главы 25 узнал кое-что новое для себя.

И наконец, люди, которые пишут выдающиеся книги, сами неординарны. Сообщество, сформировавшееся вокруг Mac и iPhone, довольно компактно, в нем все друг друга знают. Очень быстро становится известно, кому можно доверять. Многие отстойные книги, о которых я упоминал выше, написаны людьми, о которых я никогда не слышал ни до, ни после. Они не ведут блогов, не появляются на конференциях, не оставляют по себе никакого следа в сообществе. Возник из ниоткуда, что-то сфабриковал и пропал.

Кирби и Том – совсем другое дело. Это известные личности. У них есть блоги. Том – один из соавторов книги «Core Data for iOS: Developing Data-Driven Applications for the iPad, iPhone, and iPod touch». Они поставляют программные продукты. У них имеются довольные клиенты. Они отвечают на вопросы в сети. Они организуют конференции и выступают на них. Они приходят на собрания местных клубов CocoaHeads. Они тратят немало личного времени на то, чтобы сообщество стало лучше. Поэтому я расценил как большую честь предложение написать предисловие для этой книги.

Вы, наверное, поняли, что меня эта книга привела в восторг. Есть много отличных книг, посвященных введению в программирование для iOS. Я рекомендую прочесть их все (ну, по крайней мере, хорошие) потому что iOS – настолько обширная тема, что даже Кирби и Том не сумеют уместить в один том все, что необходимо знать. Но если вас интересует конкретно программирование iPad, то эта книга – как раз то, что надо. У меня есть предчувствие, что для многих она станет одним из наиболее авторитетных текстов.

– Марк Далримпл

Сооснователь CocoaHeads, международного сообщества программистов для Mac и iPhone.

Автор книги «Advanced Mac OS X Programming: The Big Nerd Ranch Guide».

12 ноября 2011



ВСТУПЛЕНИЕ

В октябре 2011 года генеральный директор Apple Тим Кук поделился интересными фактами об iPad, в частности:

- 92 процента компаний из списка Fortune 500 тестируют или внедряют приложения для «айпадов»;
- свыше 80 процентов больниц в США тестируют «айпады» или уже начали пробную эксплуатацию;
- во всех штатах США имеется та или иная программа внедрения «айпадов» – в перспективе или на этапе пробной эксплуатации.

И на этом новости об «айпадах» не заканчиваются. Федеральное управление гражданской авиации США одобрило применение «айпадов» вместо бумажных самописцев для пилотов гражданской авиации при исполнении служебных обязанностей. Без сомнения «айпады» меняют наши представления о компьютерах и их использовании. А с выходом операционной системы iOS 5 для iPad и iPhone эти устройства стали еще лучше.

Не стоит недооценивать возможности iPad. Со своим патентованным мультисенсорным интерфейсом, мощным процессором A5 и поддержкой сетей 3G и WiFi iPad является эталоном в мире после ПК. Важнее, однако, то, как iPad 2 интегрирован в экосистему Mac/iOS. Пользователи Mac OS X Lion и iOS 5 могут с помощью программы FaceTime организовывать видеочат между настольным компьютером и мобильным устройством. Более того, входящая в состав iOS 5 программа iMessage позволяет владельцам iPad и iPhone обмениваться текстовыми сообщениями. iPad, представляющий собой уникальное сочетание оборудования и технологии, задает тон для всех планшетов.

Эта книга ориентирована на операционную систему iOS 5 и рассчитана на разработчиков-новичков, желающих писать приложения для iPad. Она придется по вкусу и разработчикам для iPhone, которые хотели бы больше узнать о том, как адаптировать свои приложения для iPad. Многие считают, что iPad – это просто большой iPhone, но на

самом деле это не так. С точки зрения пользовательского интерфейса, iPad позволяет куда больше, чем iPhone.

Хотя в этой книге встречаются краткие отступления по поводу программирования iPhone, в основном она посвящена iPad. Особое внимание уделяется уникальным для iPad частям iOS 5 SDK; это не просто перекомпиляция аналогичных книг по iPhone. Кроме того, рассматриваются такие новшества iOS 5, как контроллеры контейнерных представлений, технологии iCloud и Core Image, и такие появившиеся в среде Xcode 4.2 средства, как раскадровки. Компания Apple много сделала для того, чтобы упростить разработку для iOS и OS X, а в этой книге мы хотим показать вам, как этим богатством воспользоваться.

Чему я научусь?

Из этой книги вы узнаете, как писать приложения для iPad. Шаг за шагом мы пройдем весь путь создания реального приложения, которое прямо сейчас можно бесплатно скачать из магазина App Store. Это приложение называется PhotoWheel.

Скачайте приложение!

Скачать приложение PhotoWheel из App Store можно по адресу: itunes.apple.com/app/photowheel/id424927196&mt=8. Оно бесплатно, поэтому не тяните – скачивайте и начинайте пользоваться.

PhotoWheel – это вариация на тему приложения Photos, имеющегося на любом «айпаде». Оно позволяет распределять любимые фотографии по альбомам, посылать их друзьям и родственникам по электронной почте и просматривать на подключенном по беспроводной сети телевизоре с помощью технологии AirPlay. Но самое главное то, что на его примере вы научитесь писать собственные приложения. Вы узнаете о новейших возможностях iOS 5 и Xcode, в том числе о раскадровках, автоматическом подсчете ссылок, технологиях iCloud и Core Image, а также о том, как использовать такие средства iOS, как AirPrint, AirPlay и Grand Central Dispatch (GCD). И увидите, как расширить рамки своего приложения за счет взаимодействия с веб-службами, размещенными в Интернете.

Можете рассматривать эту книгу как сверхдлинное учебное пособие, в котором описывается процесс разработки реального приложения для iPad от начала до конца. На протяжении всей книги вы будете

писать код, а мы – объяснять каждый шаг. В конце вы будете иметь полнофункциональную программу PhotoWheel, которую сможете с гордостью показывать родственникам и знакомым (можете даже поделиться с ними). И главное – будете понимать, как проектируются, кодируются и распространяются приложения для iPad.

В чем состоят особенности iPad?

Несмотря на то, что iPad работает под управлением той же операционной системы iOS, что и iPhone, iPod touch и Apple TV, он существенно отличается от других устройств. Все устройства используются по-разному, а iOS наделяет каждое из них определенной функциональностью. Например, версия iOS, работающая в Apple TV, не располагает сенсорным интерфейсом. Пользовательский интерфейс Apple TV вообще устроен совершенно иначе, он реализован в виде слоя поверх iOS и предлагает пользователю иные средства взаимодействия.

Но iPad – это нечто особенное. Его не держат на ладони, как iPhone или iPod touch, – нужны обе руки. Его экран реагирует на касания. Он понимает разные жесты, например смахивание. Можно, конечно, называть iPad «большим iPhone», но это будет неверно.

Конечно, разницу в физических размерах iPad и iPhone не заметить трудно, однако истинное различие между ними куда глубже – заключается оно в том, как проектируются приложения для iPad и как пользователь взаимодействует с ними. И начинается все с большего размера экрана.

Большой экран

Площадь экрана iPad в два с лишним раза больше, чем у iPhone. А значит, приложение может вывести больше информации в пользовательский интерфейс. Хорошим примером может служить приложение WeatherBug.

Приложение WeatherBug использует всю площадь экрана iPad. На рис. P.1 видно, что версия WeatherBug для iPad выводит на одном экране гораздо больше информации о погоде, чем версия для iPhone. Вам не нужно ничего смахивать, чтобы получить дополнительные сведения (которые еще не сразу и найдешь); все необходимое уже есть на главном экране. Разумеется, вывод детальных сведений по касанию также поддерживается.

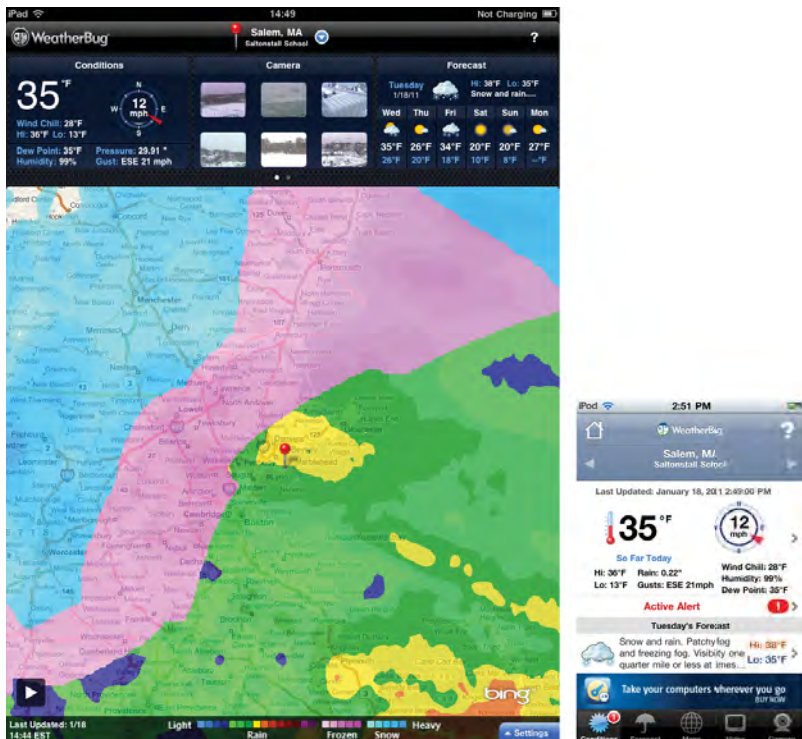


Рис. Р.1. Слева показано приложение WeatherBug для iPad, справа – то же самое приложение для iPhone. (Воспроизводится с разрешения компании Earth Networks.)

Менее глубокие иерархии

Из-за небольшого экрана во многих приложениях для iPhone применяется иерархическая система навигации. Пользователь касается какого-то объекта – появляется новый экран. Следующее касание – следующий экран. Для возврата нужно коснуться кнопки «Назад», которая обычно находится в левом верхнем углу экрана.

Хорошей иллюстрацией такой иерархической навигации может служить приложение Dropbox. Если кто не знает, Dropbox – это онлайн-сервис, позволяющий сохранять файлы, документы и изображения в облаке. При этом производится синхронизация сохраненных файлов на разных компьютерах и устройствах, на которых запущена клиентская часть Dropbox. Предположим, к примеру, что

вы поработали с текстовым документом на своем ноутбуке и сохранили его в папке Dropbox. Затем вы хотите просмотреть документ, но уже на iPhone. Благодаря Dropbox это возможно.

На iPhone Dropbox показывает отсортированный по алфавиту список файлов и папок. Когда вы касаетесь файла (или папки), тот открывается и на экране появляется новое представление. Открыв файл, вы увидите его содержимое, а, открыв папку, – новый список файлов и папок. Таким образом, касаясь папок, вы продвигаетесь вглубь иерархии.

Чтобы подняться вверх на один уровень, нужно коснуться кнопки в левом верхнем углу. Текст на этой кнопке может меняться. Обычно там показывается имя предыдущего элемента в стеке, но иногда – просто слово *Back*. При этом стиль кнопки неизменен – ее левый край заострен, так что пользователь понимает, что кнопка служит для перемещения между экранами.

Двусторонняя навигация по иерархии папок показана на рис. P.2.

Имеется также версия Dropbox для iPad. Как же разработчикам удалось перепроектировать приложение, которое очевидно нуждалось в иерархической навигации, и сделать навигацию более «плоской»? Они воспользовались объектом iOS `UISplitViewController`, который доступен только для iPad (см. рис. P.3).

Контроллер разделенного представления – это невизуальный объект, управляющий отображением двух расположенных бок о бок представлений. Если вы держите iPad в альбомной ориентации, то видны оба представления. Но стоит повернуть iPad – и левое представление исчезнет, чтобы пользователь мог сосредоточиться на основном содержимом, отображаемом справа.

Примечание. О том, как программируются приложения с разделенными представлениями, рассказывается в главе 8 «Создание приложения с интерфейсом “основной-подробности”».

Такой тип интерфейса часто называют «основной-подробности»; основное представление отображается слева, а подробное – справа. Основное представление служит для навигации по данным – в случае Dropbox по списку файлов и папок. Когда нужный файл окажется в основном представлении, вы можете коснуться его – содержимое появится в подробном представлении справа. Теперь можете повернуть iPad на 90 градусов и сосредоточиться на содержимом файла, убрав с экрана основное представление.



Рис. Р.2. Пример навигации по иерархии папок и файлов в приложении Dropbox для iPhone. Для перемещения на следующий уровень иерархии и для открытия файла нужно коснуться самого объекта, а для возврата – кнопки «Назад»

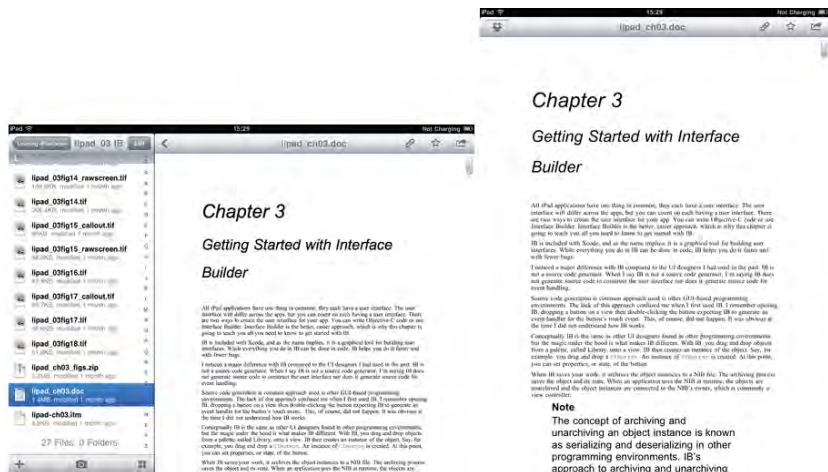


Рис. Р.3. Приложение Dropbox для iPad. Обратите внимание, что в альбомной ориентации видно левое представление с панелью навигации, а в книжной ориентации оно скрыто

Ориентация имеет значение

Большинство приложений для iPhone поддерживают только одну ориентацию. Игры как правило работают в альбомном режиме, а прочие приложения – в книжном. iPhone, как и iPad, поддерживает поворот и различные ориентации, но из-за малого размера экрана эта возможность остается невостребованной. Пользователи обычно держат iPhone в книжной ориентации, так что кнопка **Home** находится внизу, и поворачивают его, только чтобы поиграть.

С iPad всё по-другому. Пользователь iPad держит устройство, как ему удобно, не обращая особого внимания на ориентацию, особенно если нет футляра. Давайте проведем небольшой эксперимент.

Положите свой iPhone или iPod touch на стол, так чтобы кнопка **Home** указывала на 10 часов. Отойдите или отвернитесь от стола. Вернитесь и возьмите устройство в руки. Посмотрите, как вы его держите. Скорее всего, поднимая устройство, вы повернули его так, чтобы кнопка **Home** оказалась внизу. Причем вы поворачиваете его, еще даже не включив. Это почти инстинктивное движение.

Теперь проделайте то же самое с iPad. Положите устройство на стол, так чтобы кнопка **Home** указывала, скажем на 10 часов, и отойдите. Вернитесь и возьмите iPad. Почти наверняка вы не поворачивали устройство, а держите его в том же положении, в каком оно лежало на столе.

Мультисенсорность наступает

А вы знаете, что iPad и iPhone поддерживают один и тот же мультисенсорный интерфейс? Теперь знаете. На самом деле, iOS поддерживает до 11 одновременных касаний. Это значит, что для работы с приложением можно использовать пальцы обеих рук да еще привлечь приятеля.

Благодаря большому экрану задействовать возможности мультисенсорного интерфейса на iPad проще. Если на iPhone жесты двумя руками находят ограниченное применение, то при взаимодействии с приложением для iPad они выглядят вполне органично. Взять, к примеру, приложение Keynote для iPad, поставляемое самой компанией Apple. В нем мультисенсорный интерфейс используется для тех же целей, для которых в программах для настольного ПК служит наведение мышью с последующим щелчком. Так, можно одним движением выбрать и перетащить сразу несколько слайдов.

Итак, теоретически вы знаете, что мультисенсорный интерфейс поддерживает до 11 точек касания, но как в этом убедиться? Написать приложение, которое подсчитывает количество одновременных касаний. Именно так и поступил Мэтт Лидженд Джеммелл (Matt Legend Gemmell). Он написал симпатичное приложение для iPad (рис. Р.4), которое показывает количество касаний. Но этим Мэтт не ограничился. Он придал приложению фантастический вид, так что с ним интересно поиграть.



Рис. Р.4. Пример приложения с мультисенсорным интерфейсом, написанный Мэттом Лидженд Джеммелем, – показано 11 точек касания

Почитать об этом примере и скачать его исходный код можно в блоге Мэтта (mattgummell.com/2010/05/09/ipad-multi-touch). Еще один способ исследовать мультисенсорные возможности iPad – поиграть в игру Uzu для iPad, которая продается в App Store всего за \$1.99 (bit.ly/learnipadprog-UzuApp). Uzu – это «кинетический мультисенсорный визуализатор частиц», оторваться от этой игры невозможно. (Картинка на рис. P.5 не дает истинного представления о приложении; чтобы увидеть хитроумное применение мультисенсорного интерфейса на практике, его нужно скачать и попробовать.)

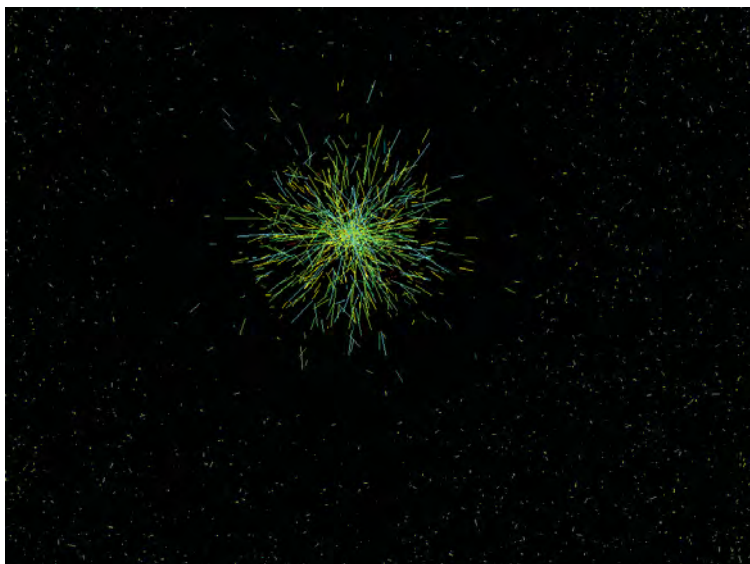


Рис. P.5. Uzu – визуализатор частиц для iPad

iPad ликвидирует разрыв между телефоном и компьютером

Итак, все согласны, что iPad – не просто увеличенный iPhone. Отлично, рад, что вы добрались до этой страницы. А теперь вопрос посложнее: можно ли считать iPad заменой ноутбуку или настольному ПК? Пока нет, но чертовски близко.

Для многих iPad служит мобильным устройством, которое заполняет брешь между смартфоном и полноценным компьютером – переносным или настольным. И хотя чаще всего iPad используется для ознакомления с контентом, его также применяют для многих задач,

которые раньше оставались вотчиной ноутбуков или настольных ПК. Это заставляет разработчиков iOS заново придумывать, как реализовать программные концепции, существовавшие «давным-давно». Одна из них – программы для обработки текстов, начавшие на iPad новую жизнь.

iPad открывает дорогу многочисленным приложениям, не уместяющимся в мелкий формат iPhone. И снова на ум приходят текстовые процессоры.

iPhone, конечно, вполне подходит для ввода коротких заметок, но для составления больших документов он пригоден мало. И хотя технически реализовать полноценный текстовый процессор для iPhone возможно, но кому это нужно? Экран слишком мал, и даже в альбомном режиме печатать двумя пальцами на крохотной клавиатуре – занятие мало продуктивное. iPhone идеален для простых и быстрых задач – написать заметку, запланировать мероприятие, пометить, что намеченное дело сделано, – но для таких длительных занятий, как написание книги, он не годится.

Здравствуй, iPad

А вот iPad позволяет почувствовать себя так, будто работаешь с маленьким ноутбуком. А в сочетании с беспроводной клавиатурой он становится вполне приемлемым средством для написания длинных документов. Это я по себе знаю. Значительная часть этой книги была первоначально написана на «айпаде». Не могу себе представить, как этот процесс выглядел бы на «айфоне», но точно знаю, что на «айпаде» я не испытывал неудобств. И, что немаловажно, iPad позволяет сосредоточиться на одной задаче, не отвлекаясь на посторонние окна.

Структура книги

Эта книга представляет собой практическое руководство по программированию iPad. В ней последовательно рассматриваются все шаги разработки, начиная от загрузки и установки iOS SDK и кончая отправкой приложения для анализа.

Книга состоит из 27 глав и одного приложения.

- Часть I. «Приступая к работе»

Часть I знакомит с инструментарием. Вы познакомитесь с такими средствами разработки, как Xcode и Interface Builder. Вы

научитесь писать код на языке Objective-C с использованием каркаса Cocoa. И узнаете, как подготовить iPad к разработке программ.

- Глава 1. «Ваше первое приложение»
Здесь мы не мешкая приступим к созданию первого приложения. В этой главе описывается пошаговая процедура создания простого, но вполне функционального приложения для iPad, которое работает в эмуляторе. Для разработки мы будем пользоваться средой Xcode, а это означает, что, хотя немножко кода написать все-таки придется, но знания языка Objective-C пока не потребуются. Цель этой главы – поскорее опробовать инструменты в работе и познакомиться с программной структурой приложения для iPad.
- Глава 2. «Знакомство с Xcode»
Xcode – это интегрированная среда разработки (IDE), применяемая для создания приложений для iPad на языке Objective-C. В этой главе описываются основные возможности Xcode, в том числе рекомендуемые настройки, наиболее употребительные комбинации клавиш и различные окна.
- Глава 3. «Знакомство с Interface Builder»
В этой главе мы познакомимся с конструктором интерфейсов Interface Builder (IB). Он позволяет создавать пользовательский интерфейс приложения вообще без программирования. Мы объясним, как работать с IB, и расскажем о многих полезных функциях. Попутно мы упомянем о типичных ошибках, которые допускают начинающие. Например, часто забывают ассоциировать событие с методом действием IBAction.
- Глава 4. «Знакомство с Objective-C»
В этой главе приводится краткий обзор языка Objective-C, на котором в основном и пишутся приложения для iPad. Мы не ставим себе целью дать полное описание языка, а ограничиваемся описанием средств, достаточных для создания первого реального приложения.
- Глава 5. «Знакомство с Cocoa»
Язык программирования – ничто без системы поддержки, а Cocoa – это впечатляющий набор подсистем и библиотек, который позволяет тратить меньше времени на программирование приложений.

- Глава 6. «Подготовка iPad»
На дороге из желтого кирпича, ведущей в удивительный мир разработки для iPad, нас подстерегают и опасности. Одна из самых страшных – необходимость иметь дело с профилями подготовки, сертификатами и регистрацией устройства для тестирования. В версии Xcode 4 в этой части есть некоторые улучшения, но до идеала все равно еще далеко. Эта глава станет вам проводником по жуткому лесу профилей, сертификатов и регистрации устройства.

- Глава 7. «Проектирование приложения»
Невозможно написать приложение, если не знаешь, что хочешь написать. В этой главе мы поделимся соображениями о том, как проектировать приложение еще до написания первой строки кода.

- Часть II. «Разработка приложения PhotoWheel»

Часть II – сердцевина этой книги. Именно здесь вы получите практический опыт создания реального приложения для iPad. И не какой-нибудь примитивной программы типа «Здравствуй, мир», а полноценного приложения PhotoWheel для управления коллекцией фотографий. Мы расскажем и об анимированном переходе представлений, и о синхронизации с iCloud, и о просмотре фотографий на экране телевизора.

- Глава 8. «Создание приложения с интерфейсом “основной-подробности”»
Разработку приложения PhotoWheel мы начнем с создания прототипа. Это даст нам возможность изучить контроллер разделенных представлений, применяемый в приложениях с интерфейсом типа «основной-подробности».
- Глава 9. «Табличные представления»
В этой главе мы освоим отображение данных в таблицах и узнаем, как можно переупорядочивать, удалять и даже редактировать данные в табличном представлении.
- Глава 10. «Представления»
В этой главе мы с головой окунемся в мир представлений. Вы узнаете, как создать нестандартное представление «колесо», в котором фотографии расположены по кругу.
- Глава 11. «Жесты»
В этой главе вы научитесь пользоваться возможностями мультисенсорного экрана iPad. Вы узнаете, как пользователь может взаимодействовать с приложением с помощью жестов.

- Глава 12. «Добавление фотографий»
Приложение PhotoWheel предназначено для работы с фотографиями, поэтому когда-то надо научиться фотографии добавлять. В этой главе вы узнаете, как забирать фотографии из библиотеки приложения Photos и как делать новые фотографии с помощью встроенной камеры.
- Глава 13. «Сохранение данных»
Приложение PhotoWheel будет бесполезно, если пользователь не сможет сохранить результаты работы. Есть много методов сохранения и извлечения данных. В этой главе вы узнаете о двух из них и научитесь работать с подсистемой Core Data.
- Глава 14. «Раскадровки в Xcode»
Раскадровка – это новый перспективный способ проектирования пользовательского интерфейса приложения. В этой главе мы применим раскадровки на практике и научимся с помощью Interface Builder добиваться больших результатов при меньшем объеме собственного кода.
- Глава 15. «Контроллеры представлений – добавим код»
Раскадровки – вещь замечательная, но их возможности все же ограничены. Чтобы приложение по-настоящему засверкало, нужно все-таки и код написать. В этой главе вы узнаете, как достичь большего с помощью контроллеров.
- Глава 16. «Конструирование главного экрана»
В этой главе мы приступим к написанию PhotoWheel. Прототип готов, и с помощью раскадровок мы заложили основы пользовательского интерфейса. Пришло время сконструировать главный экран. Заодно мы поговорим о контроллерах контейнерных представлений и создадим свое сеточное представление, которым можно будет пользоваться и в других проектах.
- Глава 17. «Создания обозревателя фотографий»
В этой главы вы научитесь использовать прокручиваемое представление для создания полноэкранного обозревателя фотографий. А также увидите, как с помощью жеста сведения-разведения пальцев можно масштабировать фотографию.
- Глава 18. «Поддержка поворота устройства»
Пользователь ожидает, что iPad будет правильно показывать картинку, в каком бы положении его ни держать. Кнопка **Home** может находиться слева, справа, внизу или вверху.

А обеспечить правильность отображения – ваша задача. Поддержке поворотов устройства и посвящена данная глава.

- Глава 19. «Печать с помощью AirPrint»
Эта глава посвящена именно тому, что написано в ее заголовке – печати из приложения с помощью технологии AirPrint.
- Глава 20. «Отправка электронной почты»
В наше время почти у каждого есть ящик в какой-нибудь почтовой системе. И все любят рассматривать фотки. Поэтому неудивительно, что пользователи PhotoWheel захотят отправлять свои фотографии друзьям и родственникам по электронной почте. А как это делается, вы узнаете из этой главы.
- Глава 21. «Веб-службы»
Добавление в PhotoWheel фотографий, которые уже хранятся в iPad, – само по себе полезное дело, но многие держат свои фотографии в другом месте. В этой главе вы научитесь искать фотографии на веб-сервере Flickr и скачивать их в приложение для iPad.
- Глава 22. «Синхронизация с iCloud»
У многих людей есть несколько iOS-устройств, и было бы здорово, если бы на любом из них программа PhotoWheel могла бы работать с одними и теми же данными. Вообще говоря, синхронизация – сложная задача, но благодаря технологии iCloud она оказывается гораздо проще, чем могло бы показаться. В этой главе мы реализуем онлайную синхронизацию фотографий и альбомов.
- Глава 23. «Создание слайд-шоу с помощью AirPlay»
Экран у iPad большой, ничего не скажешь, но что, если вам захочется показать фотографии группе людей? Не толпиться же им вокруг портативного устройства. В этой главе вы узнаете, как с помощью беспроводной сети iPad может выводить изображение на внешний экран, например, широкоформатного телевизора. А поскольку используется беспроводная технология AirPlay, то тянуть по всей комнате кабели не придется.
- Глава 24. «Создание визуальных эффектов с помощью Core Image»
Core Image – поразительно богатая подсистема для анализа и редактирования изображений. Если цветовых эффектов

и автоматического улучшения качества изображения недостаточно, то с помощью Core Data Image вы можете обнаруживать лица людей на фотографии. Мы добавим все это в PhotoWheel, снабдив удобным интерфейсом, позволяющим сначала посмотреть, что получится в результате применения эффекта, а только потом сохранить результат.

- Часть III. «Завершающие штрихи»

В последней части книги речь пойдет об отладке приложения. И – что еще важнее – о его распространении.

- Глава 25. «Отладка»

Вы уже знаете, как создать приложение для iPad, но что, если обнаружится ошибка? Эта глава посвящена отладке приложений. В ней вы познакомитесь с отладчиком GDB, научитесь расставлять точки прерывания и пользоваться при отладке звуковыми эффектами. Здесь же вы получите начальные сведения о более продвинутых методах отладки, например об использовании программы Instruments для поиска утечек памяти.

- Глава 26. «Распространение приложения»

Итак, приложение написано, отлажено и протестировано. Следующий шаг – доставить его пользователям. В этой главе мы рассмотрим различные способы распространения приложений для iPad, уделив особое внимание двум наиболее употребительным: Ad Hoc и App Store.

- Глава 27. «Послесловие»

Книга заканчивается напутствием программистам, только приступающим к написанию приложений для iPad.

- Приложение А. «Установка средств разработки»

В этом приложении мы последовательно опишем все шаги, которые нужно предпринять перед началом программирования для iPad: создание учетной записи разработчика для iOS, скачивание iOS SDK и установку на Mac средств разработки.

В книге «Основы программирования iPad» рассматривается весь цикл разработки приложения – от проектирования до отправки в магазин App Store. Попутно вы узнаете об инструментарии разработчика, языке программирования и имеющихся подсистемах. А главное – научитесь создавать полноценные приложения для iPad, которыми можно будет похвастаться перед друзьями и близкими.

На кого рассчитана эта книга

Книга рассчитана на программистов, которые ранее не работали с платформой iOS и хотели бы научиться писать приложения для iPad. Предполагается, что читатель ранее не программировал для iPad и совсем или почти ничего не знает о среде Xcode и языке программирования Objective-C. Однако же мы рассчитываем, что у читателя всё же есть опыт программирования в других средах на других языках. Тем, кто не имеет вообще никакого представления о программировании, эта книга не поможет.

Книга предназначена программистам, желающим узнать, как разрабатываются сложные приложения для iPad с использованием операционной системы iOS 5. Предполагается, что у читателя имеется компьютер Mac, на который можно установить Xcode и Interface Builder. Понадобится также учетная запись разработчика для iOS и сам iPad. Опыт программирования на языке C желателен, но необязателен, так как в книге есть глава, содержащая введение в объектно-ориентированное программирование на языке Objective-C.

Книга будет интересна и опытным разработчикам на платформе iOS, которые уже писали и распространяли через App Store приложения для iPhone и iPod touch. Если вы относитесь к этой категории, то можете пропустить начальные главы и сразу перейти к сквозным примерам.

Получение исходного кода PhotoWheel

Исходный код ко всем главам, а также полный исходный код приложения PhotoWheel, на примере которого ведется изложение, можно скачать с сопроводительного сайта книги (learnipadprogramming.com/source-code/). Но работа над PhotoWheel не стоит на месте. Программу можно совершенствовать в разных направлениях и при этом многому научиться. Актуальную версию исходного кода можно получить их хранилища github (github.com/kirbyt/PhotoWheel).

Кроме того, в сопроводительном блоге этой книги имеются дополнительные статьи и советы, в каких направлениях развивать PhotoWheel (learnipadprogramming.com/blog/).

Если у вас появятся вопросы или вы захотите сообщить об ошибке или предложить новую функцию для PhotoWheel, пишите на адрес

kirby@whitepeaksoftware.com или tph@atomicbird.com либо отправьте сообщение по адресу [@kirbyt](https://twitter.com/kirbyt) или [@atomicbird](https://twitter.com/atomicbird) в Twitter.

В книге много исходного кода и упражнений. Вы, наверное, захотите поэкспериментировать с кодом, поэтому предполагается, что у вас есть доступ к средствам разработки для Apple, например к Xcode и iOS SDK. То и другое можно скачать с сайта Apple iOS Dev Center¹.

Иллюстрации предоставил

Мэтт Маккрей (Matt McCray) – мировой парень, создавший всю графику для PhotoWheel. Если вам нужен дизайнер для следующего своего приложения, можете связаться с Мэттом по адресу matt@elucidata.net. Адрес его веб-сайта www.elucidata.net.

¹ Apple iOS Dev Center: developer.apple.com/ios.