

iOS Human Interface Guidelines



Содержание

Введение 6 / (7)*

С Первого Взгляда 6 / (7)

Отличное iOS Приложение Объединяет Платформу и Принципы Дизайна
Пользовательского Интерфейса 6 / (8)

Отличный Дизайн Приложения Начинается с Несколько Четких Определений 7 / (8)

Отличный Пользовательский Опыт Уходит Своими корнями в Ваше Внимания к
Деталям 7 / (8)

Пользователи Ожидают Найти iOS Технологии в Приложениях, Которые Они
Используют 7 / (9)

Все приложения Нуждаются в Некоторой Индивидуальной Творческой Работе 7 / (9)

Характеристики платформы 9 / (10)

Дисплей Имеет Огромное Значение, Независимо от Своего Размера 9 / (10)

Возможность Смены Ориентации Устройства 9 / (10)

Приложения Реагируют На Жесты, а не на Клики 10 / (11)

Пользователи Одновременно Взаимодействуют Только с Одним Приложением 11 / (12)

Предпочтения Доступны в Настройках 12 / (13)

Минимальная Экранная Пользовательская Помощь 12 / (14)

Большинство iOS Приложений Имеют Одно окно 12 / (14)

Два Типа Программного Обеспечения Запускаемого в iOS 13 / (14)

Safari на iOS Предоставляет Web-Интерфейс 14 / (15)

Принципы Пользовательского Интерфейса 16 / (17)

Эстетическая Целостность 16 / (17)

Логичность 16 / (17)

Прямая Манипуляция 17 / (18)

Обратная Связь 17 / (18)

Метафоры 18 / (19)

Пользовательский Контроль 18 / (19)

Стратегии Дизайна Приложения 19 / (20)

Создание Документации к Приложению 19 / (20)

1. Список Всех Функций, Которые по Вашему Мнению Понравятся
Пользователям 19 / (20)

2. Определите Кто Ваши Пользователи 19 / (21)

3. Фильтруйте Список Возможностей, Пока Определяете Аудиторию 20 / (21)

4. Не Останавливайтесь на Этом 20 / (22)

Дизайн Приложений для Устройства 21 / (22)

Обзор iOS UI Парадигм 21 / (22)

Убедитесь, что Универсальное Приложение Работает и на iPhone, и на iPad 22 / (23)

Переделайте Дизайн, Основанный на Web 22 / (23)

Привязка Индивидуализации к Задаче 23 / (24)

Прототип и Итерации 25 / (27)

Основы: Переход в iOS 27 / (28)

От Почты на Рабочем столе к Почте на iPhone 27 / (29)

От Презентации на Рабочем Столе к Презентации на iPad 29 / (31)

От Почты на iPhone к Почте на iPad 32 / (34)

От Desktop-Браузера к Safari на iOS 35 / (37)

*- в скобках приведена нумерация оригинального содержания iOS Human Interface Guidelines

Руководство по Пользовательскому Впечатлению 40 / (43)

Фокусируйтесь на Основной Задаче 40 / (43)

Выделяйте Контент, Который Нужен Пользователям 41 / (44)

Думайте 'Сверху Вниз' 41 / (45)

Дайте пользователям Логический Путь Для Следования 42 / (45)

Сделайте Использование Простым и Очевидным 42 / (45)

Используйте Направленную на Пользователя Терминологию 43 / (46)

Минимизируйте Усилия, Необходимые Пользователю для Ввода Данных 44 / (47)

Уменьшайте Файлозагрузочные Операции 44 / (47)

Предоставление Сотрудничества и Связей 45 / (48)

Не-Выделение Натроек 45 / (49)

Уместное Использование Бренда 47 / (50)

Сделайте Функцию Поиска Быстрой и Удобной 47 / (51)

Привлекайте и Информировать Хорошо Написанными Описаниями 49 / (52)

Будьте Лаконичными 49 / (52)

Используйте UI Элементы Должным Образом 49 / (52)

Обдумайте Добавление Физичности и Реализма 50 / (54)

Восхищайте Пользователей Умопомрачительной Графикой 52 / (55)

Поддержка Смены Ориентации 53 / (57)

Создавайте Цели Размером с Кончики Пальцев 55 / (59)

Используйте Едва Различимую Анимацию для Коммуникации 56 / (60)

Поддержка Необходимых Жестов 57 / (60)

Запрашивайте у Пользователей Сохранение Только в Случае Необходимости 58 / (61)

Очень Редко Создавайте Модальные Задачи и Делайте их Простыми 58 / (62)

Начинайте Сразу 59 / (63)

Всегда Будьте Готовы Остановиться 60 / (64)

Не Выходите Программным Образом 61 / (64)

При Необходимости Отобразите Пользовательское Соглашение или Отказ 61 / (65)

Для iPad: Усиление Интерактивности (Не Добавляйте Функции) 62 / (65)

Для iPad: Уменьшение Переходов с Полного Экрана 62 / (66)

Для iPad: Сдерживайте Иерархию Вашей Информации 63 / (66)

Для iPad: Обдумайте Использование Ротора для Некоторых Модальных Задач 65 / (69)

Для iPad: Переместите Контент Панели Инструментов Наверх 66 / (70)

Руководство по Использованию iOS Технологий 68 / (72)

Хранилище iCloud 68 / (72)

Многозадачность 70 / (74)

Центр Уведомлений 72 / (76)

Печать 76 / (81)

iAd Мультимедийные Объявления 77 / (82)

Быстрый Предпросмотр Документа 83 / (88)

Звук 84 / (89)

Понимайте Пользовательские Ожидания 84 / (89)

Определите Аудиоповедение Вашего Приложения 85 / (91)

Управление Прерываниями Аудио 89 / (95)

Гарнитурное Медиа Управление В Случае Необходимости 91 / (97)

Управление Голосом и Доступность 92 / (98)

Меню Редактирования 93 / (99)

Отменить и повторить 95 / (101)

Клавиатура и Окна Ввода 96 / (102)

Службы Определения Местоположения 97 / (103)

Руководство по Использованию iOS UI элементов 99 / (105)

Панели 99 / (105)

Статус Панель 99 / (105)

Панель Навигации 101 / (107)

Панель Инструментов	104 / (110)
Панель Ярлыков	105 / (111)
Контентные Окна	108 / (113)
Попвер (только для iPad)	108 / (114)
Разделительное Окно (только для iPad)	112 / (117)
Окно Таблицы	113 / (119)
Текстовое Окно	121 / (128)
Web Окно	122 / (129)
Контроллер Окна Содержимого	124 / (130)
Предупреждения, Списки Действий и Модальные Окна	125 / (131)
Предупреждение	125 / (131)
Список Действий	129 / (135)
Модальное Окно	132 / (138)
Контроллеры	135 / (141)
Индикатор Активности	135 / (141)
Пикер Даты и Времени	136 / (142)
Кнопка Раскрытия Деталей	137 / (143)
Кнопка Инфо	138 / (143)
Лейбл	138 / (144)
Индикатор Активности Сети	139 / (114)
Индикатор Страницы	139 / (145)
Пикер	140 / (146)
Окно Прогресса	141 / (147)
Прямоугольная Кнопка с Закругленными Краями	142 / (147)
Панель Возможностей	143 / (148)
Панель Поиска	143 / (149)
Сегментированный контроллер	145 / (150)
Слайдер	146 / (151)
Контроллер Пошагового Изменения Величины	147 / (152)
Переключатель	147 / (153)
Текстовое Поле	148 / (153)
Кнопки и Иконки, Предоставляемые системой	149 / (154)
Стандартные Кнопки для Использования в Панелях Инструментов и Навигации	150 / (155)
Стандартные Иконки для Использования в Панели Ярлыков	153 / (157)
Стандартные Кнопки для Использования в Строках Таблицы и Другие UI-Элементы	155 / (159)
Руководство по Созданию Индивидуализированных Иконок и Изображений	155 / (160)
Советы по проектированию Великолепной Иконки и Изображения	158 / (162)
Советы по Созданию Великолепной Работы для Retina Дисплея	159 / (163)
Советы по созданию Масштабируемого Изображения	161 / (165)
Иконки Приложения	161 / (166)
Картинки Запуска	162 / (168)
Маленькие Иконки	166 / (171)
Иконки Документа	167 / (171)
Спецификация Иконки Документа для iPhone	167 / (172)
Требования к Иконке Документа для iPad	168 / (173)
Web Clip Иконки	170 / (175)
Иконки для Панелей Навигации, Инструментов и Ярлыков	171 / (176)
Иконки Газетного Киоска	172 / (178)

Таблицы

Характеристики Платформы 9 / (10)

Таблица 1-1 Пользовательские Жесты, Используемые для Взаимодействия с iOS 10 / (11)

Руководство по Использованию iOS Технологий 68 / (72)

Таблица 6-1 Размеры стандартного Баннера 78 / (83)

Таблица 6-2 Размеры Полноэкранного Баннера 78 / (83)

Таблица 6-3 Категории Аудиосессии и Связанные с Ними Поведения 87 / (92)

Руководство по Использованию iOS UI-Элементов 99 / (105)

Таблица 7-1 Элементы Окна Таблицы 116-117 / (121)

Таблица 7-2 Стандартные Кнопки, Доступные для Панелей Инструментов и Навигации (простой стиль) 150-151 / (155)

Таблица 7-3 Окантованные Кнопки действия для Использования в Панелях Навигации и Инструментов 152 / (156)

Таблица 7-4 Стандартные Иконки для Использования в Панели Ярлыков 153 / (158)

Таблица 7-5 Стандартные Кнопки для Использования в Строках Таблицы и Других UI-Элементах 154 / (159)

Руководство по Созданию Индивидуализированных Иконок и Изображений 155 / (160)

Таблица 8-1 Индивидуализированные Иконки и Изображения 156-157 / (160)

Таблица 8-2 Максимальные Размеры Длинного Края Масштабируемой Иконки для Каждой Версии 173 / (179)

Введение

IOS Human Interface Guidelines описывает руководства и принципы, которые помогут вам создать превосходный и удобный в использовании пользовательский интерфейс для вашего iOS-приложения.



IOS Human Interface Guidelines не описывает, как реализовать свои проекты в коде. Когда вы будете готовы к кодированию, начните с чтения iOS App Programming Guide.

С Первого Взгляда

Работая с конвенцией платформ, у вас появится гораздо больше возможностей для создания выдающихся iOS-приложений.

Отличное iOS Приложение Объединяет Платформу и Принципы Дизайна Пользовательского Интерфейса

Пользователи ценят iOS-приложения, которые, кажется, были созданы специально для данного устройства. Например, если приложение хорошо смотрится на экране устройства и реагирует на жесты, которые пользователи понимают - оно соответствует пользовательскому ожиданию. И хотя пользователи могут не знать об основных руководящих принципах пользовательского интерфейса, таких как прямая манипуляция или последовательность, они могут сказать, когда приложение слушается их, а когда нет. Как только вы начнете проектирование iOS-приложения, убедитесь, что поняли и узнали, как включить руководящие принципы пользовательского интерфейса в дизайн, и вы можете предоставить то, что пользователи оценят.

Соответствующие главы “Platform Characteristics” (стр. 9) и “Human Interface Principles” (стр. 16)

Отличный Дизайн Приложения Начинается с Несколько Четких Определений

Когда вы обдумываете идею приложения, важно решить, какие именно функции вы собираетесь предоставить и кому. После того, как вы определили это, необходимо убедиться, что вы адаптировали внешний вид вашего приложения для устройства, на котором оно работает и задачу, которую оно решает.

Если вы переносите существующее программное обеспечение на iOS, вы столкнетесь с похожими проблемами. Как изменить дизайн существующего программного обеспечения для iOS, поможет узнать iOS Human Interface Guidelines, рассказав некоторые дизайн-решения, которые демонстрируют другие успешные переносы между устройствами, например такие, как Mail и Keynote

Соответствующие главы “App Design Strategies” (стр. 19) и “Case Studies: Transitioning to iOS” (стр. 27)

Отличный Пользовательский Опыт Уходит Своими корнями в Ваше Внимания к Деталю

Очень важно держать пользовательский опыт в голове на первом месте, при проектировании каждого аспекта вашего приложения, от того, как вы приступаете к задаче до запуска и установки вашего приложения и использования кнопок. Раскройте принципы, которые влияют на внешний вид и работу вашего приложения, в вопросах как общих, так и конкретных.

Соответствующие главы “User Experience Guidelines” (стр. 40) and “iOS UI Element Usage Guidelines” (стр. 99)

Пользователи Ожидает Найти iOS Технологии в Приложениях, Которые Они Используют

iOS обеспечивает множество отличных технологий, которые увеличивают ценность приложений, такие как многозадачность, печать и VoiceOver. При использовании iOS-устройств, пользователи могут просматривать эти технологии, как автоматически доступные, но их интеграция в приложение может потребовать усилий с вашей стороны. Если iOS технология реализована в вашем приложении, убедитесь, что следуете рекомендациям, которые управляют ее использованием.

Соответствующие глава “iOS Technology Usage Guidelines” (стр. 68)

Все приложения Нуждаются в Некоторой Индивидуальной Творческой Работе

Даже если ваше приложение решает серьезные, продуктивные задачи и использует только стандартные элементы пользовательского интерфейса, вы все равно должны обеспечить красивую иконку приложения, которой пользователи будут наслаждаться, видя ее в AppStore и на экранах. Независимо от того, сколько ваше приложение включает в себя индивидуального дизайна или не много, вам нужно знать, какие иконки и изображения требуются, и как их создавать соответствующим образом. Кроме того, если вы разрабатываете дизайн для Retina экрана, вы можете узнать некоторые методы, которые могут облегчить этот процесс.

Соответствующие глава “Custom Icon and Image Creation Guidelines” (стр. 155)

Platform Characteristics

Характеристики Платформы

Устройства, основанные на iOS, имеют ряд уникальных особенностей, влияющих на взаимодействие пользователя и приложения, с которым он работает. Наиболее успешные приложения охватывают эти особенности и обеспечивают взаимодействие пользователя и интегрированы приложений в устройстве, с которым они работают.

The Display Is Paramount, Regardless of Its Size (Дисплей Имеет Огромное Значение, Независимо от Своего Размера)

Дисплей iOS-устройств, лежит в основе пользовательского опыта. Кроме того, что люди видят красивый текст, графику и медиа на дисплее, они взаимодействуют физически, с помощью Multi-Touch интерфейса, руководствуясь своим знанием (даже когда не могут видеть экран).

Несмотря на то, что размеры дисплеев разных диагоналей и разрешений могут оказывать различное влияние на взаимодействие пользователя с приложением, некоторые действия применяются ко всем iOS-устройствам:

- максимально комфортный размер для нажатия UI элементов - 44x44 пикселя;
- отображение художественной работы в приложении очень четкое;
- Пользователи фокусируются на контенте.

Примечание: Pixel - подходящая единица измерения размера экрана устройства или размера значка, который вы создаете в графическом редакторе. **Point** - соответствующая единица измерения используемая для измерения области, отображаемой на экране. На стандартном разрешении экрана устройства, один Point равен одному Pixel, но в других разрешениях, возможны другие соотношения: на Retina-дисплее, например, один Point соответствует двум Pixel.

См. "Points Versus Pixels" в iOS Application Programming Guide для подробного изучения этой концепции.

Device Orientation Can Change (Возможность Смены Ориентации Устройства)

Пользователи могут вращать iOS-устройство в любой момент и по разным причинам. Например, иногда некоторым пользователям удобнее работать в вертикальной ориентации, а иногда - в горизонтальной. Каковы бы ни были причины для вращения устройства, пользователи ожидают от приложения основных функциональных возможностей.

Пользователи как правило запускают приложения с главного экрана и ожидают, что они запустятся в той же ориентации. В iPhone и iPad разница в отображении Home Screen, по-разному влияет на приложения:

- На iPhone и iPod Home screen отображается только в одной ориентации - вертикальной -

кнопка "Home" в нижней части. Это позволяет пользователям ожидать от iPhone загрузки приложения в той же ориентации по умолчанию.

- На iPad Home screen отображается во всех ориентациях, поэтому пользователи ожидают, что iPad-приложения запустятся в той же ориентации, в которой они находятся в данный момент.

Apps Respond to Gestures, Not Clicks (Приложения Реагируют на Жесты, а не на Клики)

Пользователи делают специфические движения пальцами, называемые жестами, для управления уникальным Multi-Touch интерфейсом iOS-устройства. Например, пользователи нажимают на кнопки для активации, нажимают или перетаскивают для просмотра длинного списка или растягивают для масштабирования изображения.

Multi-Touch интерфейс дает пользователям чувство непосредственной связи с их устройствами и помогает взаимодействовать с объектами на экране.

Пользователям комфортнее со стандартными действиями, поскольку встроенные приложения используют их постоянно. Их опыт использования встроенных приложений, дает набор жестов, подразумевающих возможность успешного использования в большинстве других приложений.

Таблица 1-1 Пользовательские жесты, используемые для взаимодействия с iOS-устройствами

Жест	Действие
Нажатие (Tap)	Нажать на контроллер или на какой-нибудь элемент (по аналогии с одним кликом мыши)
Перетаскивание (Drag)	Нажать и перетащить (для перемещения из стороны в сторону).
Резкое движение (Flick)	Быстрая прокрутка
Сильное нажатие со сдвигом (Swipe)	В строке окна таблицы для открытия кнопки Delete
Двойное нажатие (Double Tap)	Приближение в центр блока контента или изображения; Отдаление (если было приближено)
Разведение двух пальцев в сторону (Pinch open)	Приближение
Сведение двух пальцев (Pinch close)	Отдаление
Прикосновение и зажатие (Touch and hold)	В редактируемом тексте для отображения «увеличительного стекла» (magnified view) и позиционирования курсора
Встряска (Shake)	Инициация повтора или отмены действия

iPhone и iPad поддерживают multifinger-жесты. Даже если большой экран предоставляет больше возможностей для дополнительных прикосновений, это не означает, что multifinger жесты будут работать лучше.

People Interact with One App at a Time

(Пользователи Одновременно Взаимодействуют Только с Одним Приложением)

Во использования приложением в определенный момент времени, только это одно приложение отображается на переднем плане. Когда люди переключаются с одного приложения на другое, предыдущее приложение завершает работу и его пользовательский интерфейс исчезает.

До iOS 4 это означало, что при выходе приложение немедленно удалялось из памяти. В iOS 4 и выше, покинутое приложение сворачивается, и либо продолжает работать, либо нет. Эта функция называется **многозадачностью** и позволяет приложениям быть свернутыми, пока с ними не работают или пока их не закрыли.

Примечание: Многозадачность работает на некоторых устройствах под управлением iOS 4 и выше.

Большинство приложений приостанавливаются, когда их сворачивают. Свернутые приложения отображаются в многозадачном пользовательском интерфейсе, обеспечивающем эффективный способ доступа к недавно используемым приложениям. Многозадачный пользовательский интерфейс отображает свернутые приложения в нижней части экрана Home screen (показанные приложения ниже настройки iPhone).



Когда пользователи запускают свернутые приложения, они могут мгновенно возобновить просмотр от точки, где пользователи остановились, без необходимости перезагрузки пользовательского интерфейса приложения.

Некоторым приложениям, возможно, необходимо продолжать работу в фоновом режиме, пока пользователь запускает другое приложение на переднем плане. Например, пользователям может понадобиться приложение, которое воспроизводит медиа контент и

будет продолжать его воспроизведение, в то время как пользователи используют другие приложения, к примеру, проверяют свой календарь или электронную почту.

Чтобы узнать, как правильно обращаться с многозадачностью, см. "Multitasking" (стр. 70).

Preferences Are Available in Settings (Предпочтения доступны в Настройках)

Пользователи устанавливают определенные предпочтения iOS приложений, встроенных в приложение Настройки. Им необходимо выйти из текущего приложения, когда они хотят получить доступ к этим предпочтениям в настройках.

Предпочтения в приложении Настройки имеют "единожды установленный и редко изменяемый" тип. Хотя некоторые встроенные приложения имеют этот тип свойств в приложении Настройки, большинство приложений не нуждаются в этом, поэтому они не имеют свойств в приложении Настройки.

Onscreen User Help Is Minimal (Минимальная Экранная Пользовательская Помощь)

Мобильные пользователи не имеют ни времени, ни желания, чтобы читать массу контента-справки, прежде чем смогут извлечь пользу из приложения. Более того, содержимое справки занимает ценное место в памяти для хранения и отображения.

iOS-устройства и их встроенные приложения интуитивно понятны и просты в использовании, так что люди не нуждаются в экранных справочных материалах, направленных на объяснения использования устройства или приложения. Этот опыт заставляет людей ожидать, что все iOS приложения аналогичны в простоте использования.

Most iOS Apps Have a Single Window (Большинство iOS Приложений Имеют Одно Окно)

iOS приложение имеет одно окно, если не поддерживает внешний дисплей. Окно приложения заполняет главный экран устройства и предоставляет пустые поверхности, которые располагаются в одном или нескольких видах (views), в которых вы предоставляете ваш контент. Важно понимать, что окно в iOS приложении очень отличается от окна в компьютерном приложении. Например, окно в iOS имеет невидимые компоненты (такие как заголовок окна или кнопки закрытия), и они не могут быть перемещены на новое место на экране устройства.

Важно также понимать, что большинство пользователей не понимают разницы между окном и отображением в iOS приложениях, которые они используют. По большей части, пользователи воспринимают приложение, как набор экранов, через которые они взаимодействуют с ним. С этой точки зрения, экран в целом соответствует четким визуальным областям или режимам в приложении. В приложении Contacts на iPhone, например, пользователи думают о своем списке контактов (вне зависимости от его длины) в качестве одного экрана и о деталях контакта – в качестве другого.

Примечание: Потому что этот документ фокусируется на пользовательский интерфейс и удобство работы с iOS-приложениями, в нем часто используется "экран" ("screen"), как его

понимает большинство пользователей. Если не указано обратное, в этом документе не используется “экран” для обозначения объекта UIScreen (который представляет собой объект, который можно использовать для доступа к экрану стеклянного дисплея).

Two Types of Software Run in iOS (Два Типа Программного Обеспечения Запускаемого в iOS)

Есть два типа программного обеспечения, которое можно разрабатывать для iOS-устройств:

- iOS-приложения;
- Web-контент.

iOS приложение - приложение, которое вы разрабатываете с использованием iOS SDK, чтобы оно работало, как родное для iOS-устройств. iOS-приложения похожи на встроенные приложения для iOS-устройств в том, что они находятся на самом устройстве и используют преимущества среды iOS. Пользователи устанавливают iOS приложения на свои устройства и используют их так же, как они используют встроенные приложения, такие как фотографии, календарь и почта.

Web-контент сайта размещенного на сервере, люди просматривают через iOS-устройства. Web-контент, который появляется на iOS-устройствах, может разделяться на три категории:

Web-приложения. Web-страницы, которые обеспечивают выполнение задачи и соответствуют определенным требованиям дисплея, известны как web-приложения, потому что они ведут себя подобно iOS приложениям.

Web-приложение часто скрывает пользовательский интерфейс Safari на iOS, чтобы он был больше похож на родное приложение. С помощью функции web-клип, веб-приложение может также предложить иконку для пользователей, для использования на Home screen. Это позволяет пользователям запускать web приложения так же как они запускают iOS приложения.

Оптимизированные Web-страницы. Web-страницы, которые оптимизированы для Safari на iOS дисплее и работают как было задумано (за исключением каких-либо элементов, основанных на неподдерживаемых технологиях, таких как плагины, Flash и Java). Кроме того, оптимизированные web-страницы правильно масштабируют контент для экрана устройства и часто предназначены для распознавания при просмотре на iOS устройстве, так что она может настроить содержание и обеспечить соответствие.

Совместимые Web-страницы. Веб-страницы - которые совместимы с Safari на iOS дисплее и работают, как задумывалось. Совместимая web-страница не стремится предпринять дополнительные шаги, чтобы оптимизировать качество просмотра на iOS-устройствах, но обычно устройства отображают страницы успешно.

iOS-приложения могут комбинировать родные элементы пользовательского интерфейса для доступа к web-контенту через окно просмотра web-контента. Такое приложение может выглядеть и вести себя, как родное iOS-приложение, не привлекая внимание к тому факту, что оно зависит от web-источников.

Safari on iOS Provides the Web Interface (Safari на iOS предоставляет Web-Интерфейс)

Safari на iOS предоставляет интерфейс для просмотра web-контента на iOS-устройствах. Хотя Safari на iOS во многом похожа на Safari на рабочем столе компьютера, это не одно и то же.

По большому счету, пользователи не смогут изменить **размер окна** (область, отображающую содержимое). На рабочем столе пользователь изменяет размер окна, изменяя размер окна браузера. На iOS-устройствах окно просмотра не изменяется при изменении ориентации устройства. iOS пользователи могут изменять размер окна просмотра с помощью масштабирования и перемещать web-страницы. На iPad пользователи имеют гораздо меньше возможностей для увеличения web-контента, нежели на iPhone. (см. ниже)



Safari на iOS поддерживает cookies. Использование cookies могут упростить взаимодействие пользователя с web-контентом за счет сохранения пользовательского контента, предпочтений и ранее введенных данных.

Safari на iOS не поддерживает Flash, Java (включая Java-апплеты), либо сторонние плагины в web-контенте.

Вместо этого, Safari на iOS поддерживает HTML5 <audio> и <video> тэги для предоставления аудио и видео потока, а так же JavaScript и CSS3 переходы и анимации для отображения анимированного контента.

Сафари на iOS интерпретирует большинство жестов как способ ориентации в устройстве для отображения контента, а не самого контента. Нажатие аналогично одному щелчку мыши, может привести к тому, что Safari на iOS отправит *события OnClick* на web-страницу. Аналогов другим движениям мыши нет.

Safari на iOS позволяет web-приложениям работать в полноэкранном режиме. Web-приложения, которые запускаются через иконку web-клипа с Home screen, могут скрыть пользовательский интерфейс для Safari на iOS, чтобы они больше походили на родные приложения.

Human Interface Principles

Принципы Пользовательского Интерфейса

Отличный пользовательский интерфейс, который следует общечеловеческим принципам дизайна интерфейса, основан на логике: от людей к пользователям и от пользователей к мысли и работе, а не на свойствах устройства. Непривлекательный, запутанный и нелогичный пользовательский интерфейс может сделать даже отличное приложение рутинным в использовании. Но красивый, интуитивно понятный пользовательский интерфейс убедительно повышает функциональность приложения и вдохновляет пользователей на положительную эмоциональную привязанность.

Aesthetic Integrity (Эстетическая Целостность)

Эстетическая целостность - не показатель красоты приложения. Это мера того, насколько хорошо внешний вид приложения интегрируется с его функцией. Например: приложение, которое позволяет продуктивно решать задачу, в целом сохраняет декоративные элементы в скрытом или в свернутом режиме, уделяя при этом внимание задаче, предоставляя стандартные элементы управления и жесты. Такое приложение очень ясно дает понять пользователю о своем предназначении и своеобразности. Если, с другой стороны, приложение позволяет продуктивно решать задачу в пользовательском интерфейсе, который кажется причудливым или легкомысленным, пользователи могут не знать, как интерпретировать эти противоречивые сигналы.

Кроме того, от приложений, которые погружают с головой в задачу, таких как игры, пользователи ожидают красивый интерфейс, предполагающий развлечения и открытия. Хотя пользователи не ожидают выполнения серьезной или продуктивной задачи в игре, они ожидают синтезирования игры с предоставлением уникального опыта.

Consistency (Логичность)

Логичность интерфейса позволяет использовать пользовательский опыт в различных приложениях. Логика приложения не должна повторять другие приложения. Скорее, это приложение, использующее стандарты и парадигмы, комфортные для пользователей.

Для того, чтобы приложение следовало принципу логичности, следует помнить об этих вопросах:

- Соответствует ли приложение стандартам iOS? Использует ли оно предоставляемые системой управление, окна и иконки правильно? Надежно ли используются функции устройства, включенные в приложение?

- Последовательно ли приложение? Используется ли единая терминология и стиль в тексте? Соответствуют ли значки их смыслу? Могут ли пользователи предсказать, что произойдет при выполнении однотипных действий? Есть ли UI элементы, идентичные в отображении и поведении?

- Соответствует ли приложение, в пределах разумного, своим прошлым версиям? Остались ли элементы и смысл прежними? Существенно ли изменились фундаментальные концепции?

Direct Manipulation (Прямая манипуляция)

Когда пользователи напрямую манипулируют экранными объектами, вместо использования отдельных контроллеров для манипуляции, они больше сфокусированы на задаче, и им более понятны их действия. iOS пользователи наслаждаются повышенной чувствительностью непосредственной работы с Multi-Touch интерфейсом. Использование жестов позволяет пользователям более чувственно ощутить контроль над объектами, которые они видят на экране, поскольку они имеют возможность прикоснуться к ним, не используя посредников, таких как мыши.

Например, вместо нажатия на элементы управления масштабированием, пользователи могут использовать жесты пощипывания (pinch), чтобы непосредственно увеличивать или уменьшать все, что касается содержания. А в игре пользователи взаимодействуют непосредственно с экранными объектами. Например, в игре может появиться кодовый замок, который пользователи смогут вращать для его открытия.

В iOS-приложении пользователи могут ощущать прямое манипулирование, когда они:

- Влияют на элементы экрана при помощи поворота или любого другого вращения устройства
- Манипулируют экранными объектами, используя жесты
- Наблюдают, что их действия имеют непосредственные, видимые результаты.

Feedback (Обратная Связь)

Обратная связь подтверждает действия пользователей и заверяет их, что процесс происходит. Пользователи ожидают немедленного отклика, когда они оказывают воздействие, и ценят обновление статуса во время длительных операций.

Встроенные iOS приложения отвечают на каждое действие пользователя некоторыми заметными изменениями. Например, элементы списка кратковременно подсвечиваются, когда пользователи касаются их.

Во время операций, которые длятся более нескольких секунд, дисплей приложения отображает прогресс, а в случае необходимости, выводит поясняющее сообщение.

Едва различимая анимация может дать пользователям эффективную обратную связь, которая поможет уточнить результаты их действий. Например, списки могут анимировать добавление новой строки, чтобы помочь людям визуально отслеживать изменения.

Звук также может дать пользователям полезную информацию. Но звук не должен быть главным или единственным механизмом обратной связи, потому что люди могут использовать свои устройства в местах, где они не могут слышать или приходится выключать звук.

Metaphors (Метафоры)

Когда виртуальные объекты и действия в приложении соответствуют метафорам для объектов и действий реального мира, пользователям проще понять, как использовать приложение. Классический пример метафоры - папки программного обеспечения: пользователи помещают вещи в папки в реальном мире, поэтому они сразу понимают предложение о переносе файлов в папки на компьютере.

Наиболее подходящие метафоры предлагают использование или получение опыта без вынужденных ограничений реальных объектов или действий, на которых они основаны. Например, пользователи могут разместить в папках программного обеспечения гораздо больше контента, чем поместится в физическую папку.

iOS предоставляет большой простор для метафор, поскольку он поддерживает разнообразие графических изображений и жестов. Пользователи физически взаимодействуют с реалистичными объектами экрана, во многих случаях задействуя их, как если бы они были реальными объектами. Метафоры в iOS включают в себя:

- Нажатие на контроллеры воспроизведения медиаконтента в iPod
- Перетаскивание, пролистывание или прокручивание объектов в игре
- Раздвижные On / Off переключатели
- Перелистывание страниц фотографии
- Прокручивание колеса выбора, чтобы сделать выбор

В общем, метафоры работают лучше, когда они не многозначны. Например, удобство папок программного обеспечения может снижаться, если организовать их в структуру виртуальных файлов.

User Control (Пользовательский Контроль)

Пользователи, а не приложения, должны инициировать и контролировать действия. Хотя приложение может предложить варианты действий или предупредить об опасных последствиях; распространенная ошибка при разработке приложения – предоставление пользователю возможности выбора. В лучших приложениях найден правильный баланс между предоставлением пользователям необходимых возможностей и помощью во избежание нежелательных результатов.

Пользователям комфортнее контролировать приложения, когда поведение и управление знакомы и предсказуемы. И, когда действия просты и понятны, пользователи легко могут понять и запомнить их.

Пользователи хотят иметь возможность отменить операцию до ее начала, чтобы получить возможность подтверждения своего намерения совершить потенциально разрушительные действия. Наконец, люди ожидают возможности корректной остановки операции на стадии реализации.

App Design Strategies

Стратегии Дизайна Приложения

Все больше приложений начинаются с большой идеи, но это не значит, что путь от идеи до успешного iOS приложения будет легким. В этой главе описываются некоторые стратегии, которые можно использовать для усовершенствования вашей идеи, пересмотра вариантов дизайна и помощи в реализации приложения, которое будет оценено пользователями.

Create an Application Definition Statement (Создание Документации К Приложению)

Документация к приложению формулируется кратко, конкретизируя основную идею приложения и его целевую аудиторию.

Создание документации к приложению в начале проектной разработки поможет Вам превратить идею и список функций в единый продукт, востребованный пользователями. Используйте документацию во время проектной разработки для определения надобности потенциальных функций и возможностей приложения. Выполните следующие шаги для создания хорошей документации к приложению.

1. Список Всех Функций, Которые по Вашему Мнению Понравятся Пользователям

Иди и продумай все. На данный момент вы пытаетесь охватить все задачи, связанные с вашей основной идеей продукта. Не переживайте, если ваш список окажется очень большим, вы сможете уменьшить его позже.

Например, представьте, что ваша первоначальная идея заключается в разработке приложения, которое будет помогать пользователям делать покупки в магазине. При изучении данной деятельности вы столкнетесь со списком задач (потенциальных возможностей), которые могут заинтересовать пользователей, таких как:

- Создание списков;
- Получение рецепта;
- Сравнение цен;
- Доступность в магазинах;
- Добавление заметок к рецептам;
- Получение и использование купонов;
- Просмотр демо-приготовления;
- Изучение различных кухонь;
- Поиск замены ингредиента.

2. Определите, Кто Ваши Пользователи

Кроме того, что Ваши пользователи индивидуальны во мнениях и ожидают красивую графику, простое взаимодействие и высокую производительность, что их объединяет? В рамках приложения, которое вы планируете, что является наиболее важным для пользователей? Используя покупку продуктов, к примеру, вам требуется спросить ваших пользователей:

- Обычно они готовят дома или предпочитают готовые блюда?
- Признают ли пользователи купоны или думают, что они не стоят усилий?
- Наслаждаются поиском специальных ингредиентов или редко выходят за рамки основных?
- Следуют ли рецептам или используют их, как вдохновение?
- Чаще покупают небольшое количество продуктов или редко, но оптом?
- Хотят быть в курсе нескольких незавершенных списков для различных целей или просто хотят запомнить несколько вещей, чтобы купить по дороге домой?
- Предпочитают конкретные бренды или обходятся наиболее удобными альтернативами?
- Как правило, покупают похожий набор элементов за каждый поход в магазин или покупают, согласно пунктам, перечисленным в рецепте?

После размышления над этими вопросами представьте, что вы выделили три характеристики, которые лучше всего характеризуют вашу целевую аудиторию: Любовь экспериментировать с рецептами, частая ограниченность во времени, и экономия (если это не займет слишком много усилий).

3. Фильтруйте Список Возможностей, Пока Определяете Аудиторию

Если после принятия решения по нескольким характеристикам аудитории, вы в конечном итоге имеете несколько функций, вы на правильном пути: хорошие iOS приложения фокусируются на задаче, которую хотят решить пользователи.

Например, рассмотрим большое количество возможных функций для приложения покупки товаров, которые вы перечисляете в шаге 1. Хотя все эти функции полезны, не может быть, что бы каждая функция была одинаково полезна каждому пользователю. Более важно узнать, будет ли одинаково оценена пользователями каждая функция, определенная вами в шаге 2.

Когда вы изучите ваш список возможностей в контексте вашей целевой аудитории, вы поймете, что приложение должно быть основано на трех основных функциях: создание списков, получение и использование купонов, а также получение рецептов.

Теперь вы сможете создавать документацию к приложению конкретно и кратко, о том, что приложение делает и для кого. Хорошим слоганом этого приложения может быть:

"Создание списка покупок - инструмент для экономных людей, которые любят готовить."

4. Не Останавливайтесь На Этом

Используйте свою документацию к приложению на протяжении всего процесса разработки, для определения пригодности функций, элементов управления и терминологии. Например:

Когда вы считаете, что надо добавить новую функцию, подумайте, соответствует ли это основной цели вашего приложения и вашей целевой аудитории. Если это не так, отложите ее в сторону, это может послужить основой другого приложения. Например, вы решили, что пользователи заинтересованы в активном приготовлении пищи, поэтому сфокусировались на тортах и готовых блюдах, которые, вероятно, не будут оценены.

При рассмотрении внешнего вида и поведения UI, спросите себя, оценят ли пользователи простоту стиля, его рациональность или более тематическую направленность.

Руководствуйтесь тем, что пользователи ожидают достичь с помощью вашего приложения: выполнять серьезные задачи, получать быстрые ответы, копаться в обширном содержании или развлекаться. Например, хотя ваше приложение списка покупок должно быть понятным и быстрым в использовании, ваша аудитория, скорее всего, оценит тематический UI, который включает в себя красивые фотографии ингредиентов и еды.

Как только вы определились с терминологией, старайтесь, чтобы она соответствовала пониманию аудитории.

Например, даже если ваша аудитория не может состоять из профессиональных поваров, вы достаточно уверены, что они ценят приложение, видя соответствующие определения для ингредиентов и технологий.

Design the App for the Device (Дизайн Приложений Для Устройства)

Вы знаете, что ваше приложение делает и кто его аудитория, а сейчас вам нужно убедиться, что ваше приложение выглядит и ощущается так, как должно было бы при проектировании специально для iOS-устройства. Это очень важно, потому что люди возлагают большие надежды на приложение, раз они решили установить его на свои устройства. Если дизайн вашего приложения разработан под другой тип устройства или под web, пользователи с меньшей вероятностью оценят его.

Обзор iOS UI Парадигм

iOS пользователи привыкли к внешнему виду и поведению встроенных приложений, поэтому они склонны ожидать аналогичного опыта в приложениях, которые они скачивают. Вы не хотите имитировать все детали встроенных приложений, хотя это и полезно для понимания дизайн-шаблонов, которым они следуют. Начните с понимания характеристик, которые отличают iOS устройства и приложения, работающие на них (вы можете прочитать об этом в "Platform Characteristics" (стр. 9)). Затем, соблюдайте следующие вещи:

Контроллеры должны выглядеть так, словно их можно нажать. iOS-контроллеры, такие как кнопки, пикеры и слайдеры, имеют контуры и градиенты, приглашающие взаимодействовать.

Структура приложения должна быть четкой и простой в навигации. iOS обеспечивает навигационную панель для углубления вниз по иерархическому содержанию и панель вкладок для отображения различных одноуровневых групп контента или функциональности.

Обратное взаимодействие пользователей должно быть едва заметным, но ясным. iOS приложения часто используют четкие, подвижные анимации, чтобы показать результаты действия пользователя. iOS приложения могут также использовать индикатор активности и прогресс просмотра, чтобы показать статус, и предупреждение, чтобы дать пользователям предостережения или другую важную информацию.

Вы можете узнать об использовании нормативов управления этими элементами в "iOS UI Element Usage Guidelines" (стр. 99). Когда вы приступите к разработке общего

пользовательского восприятия, убедитесь, что вы понимаете рекомендации, приведенные в “User Experience Guidelines”(стр. 40).

Убедитесь, Что Универсальное Приложение Работает, и на iPhone и на iPad.

Если вы планируете развивать приложение, которое работает на iPhone и iPad, вам нужно адаптировать свой дизайн для каждого устройства. Вот несколько способов, которые смогут помочь в этом:

Формируйте UI для каждой версии приложения, работающего на устройстве.

Большинство индивидуальных элементов пользовательского интерфейса доступны на обоих устройствах, но в целом их схема сильно отличается.

Адаптируйте арт-дизайн под размер экрана. Пользователи склонны ожидать более высокой проработки дизайна в iPad приложениях, чем в iPhone приложениях. Стандартное разрешение iPhone приложения не рекомендовано использовать на экранах iPad.

Сохраняйте основные функциональные возможности вашего приложения независимо от устройства, на котором оно работает. Хотя в одной из версий предлагается более глубокое или интерактивное решение задачи, чем других, важно, чтобы пользователям не казалось, что они выбирают между двумя совершенно разными приложениями.

Выходите за рамки, поставленные в предыдущих версиях. Немодифицированные iPhone-приложения работают в режиме совместимости с iPad по умолчанию. Хотя этот режим позволяет людям использовать iPhone-приложение на iPad, это не дает им полного опыта, который они ожидают получить.

Переделайте Дизайн, Основанный на Web

Если вы разрабатываете на основе web, вы должны убедиться, что вы даете людям ощущение iOS приложения, а не ощущение от web. Помните, что пользователи могут посетить ваш сайт через свои iOS-устройства с помощью Safari на iOS.

Здесь приведены некоторые стратегии, которые могут помочь web-разработчикам создавать iOS-приложения:

Сосредоточьтесь на своем приложении. Web-сайты часто приветствуют посетителей большим числом задач и вариантов, из которых они могут выбирать, но этот тип взаимодействия не означает, что iOS приложение хорошее. iOS-пользователи ожидают, что приложение будет делать то, что рекламирует, и хотят видеть полезную информацию немедленно.

Убедитесь, что ваше приложение позволяет людям делать что-то. Пользователи могут получать удовольствие от просмотра маркетинговых материалов на сайтах, которые они посещают, но они ожидают каких-либо действий в приложении.

Дизайн для прикосновения. Не пытайтесь повторить web-парадигмы UI-дизайна в вашем iOS-приложении.

Вместо этого ознакомьтесь с UI-элементами и шаблонами iOS и используйте их для демонстрации своего содержания. Web-элементы, которые необходимо будет пересмотреть, включают в себя: меню и взаимодействия, иницирующиеся наведением курсора, ссылки.

Позвольте людям прокрутку. Большинство web-сайтов заботятся о том, чтобы отобразить самую важную информацию в верхней части страницы, так как пользователи иногда покидают страницы, не найдя то, что они ищут, в верхней части. Но на iOS-устройствах прокрутка легка и является ожидаемой частью взаимодействия.

Если уменьшить размер шрифта или сжать контроллеры в соответствии с Вашим содержанием в пространстве одного экрана устройства, вы, вероятно, столкнетесь с нечитаемым содержанием и непригодным для использования шаблоном.

Предоставляйте значок домашней страницы. Сайты часто отображают иконку-ссылку на главную страницу в верхней части каждой страницы сайта. iOS приложения не включают домашних страниц, так что это действие не является необходимым. Кроме того, iOS приложения позволяют людям нажимать на строку состояния для быстрой прокрутки назад, в начало длинного списка. Если разместить иконку Home в верхней части экрана, это будет очень сложно для пользователей, вместо этого используйте нажатие на строку состояния.

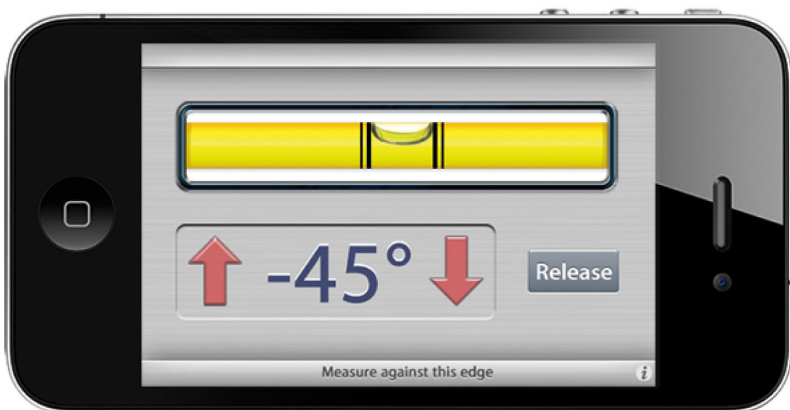
Tailor Customization to the Task (Привязка Индивидуализации к Задаче)

В лучших iOS-приложениях при индивидуализации UI используют баланс между ясностью цели и простотой в использовании. Для достижения этого баланса в вашем приложении, не забывайте об ограничениях индивидуализации в начале процесса проектирования. Потому что озабоченность по поводу брендинга, оригинальности и маркетинговой ценности часто влияет на решения в индивидуализации. Не так просто оставаться сосредоточенным больше на пользовательском восприятии, чем на индивидуализации.

Используя iOS SDK, вы можете сделать уникальное UI для вашего приложения, более или менее по своему выбору. Поскольку практически нет никаких ограничений на количество индивидуальных изменений, которые можно сделать, вы должны определить, как это может повлиять на задачи приложения. Когда вы продумываете задачи приложения, подумайте, как часто пользователи выполняют их и при каких обстоятельствах. Например, представьте приложение, которое позволяет осуществлять звонки. А теперь представьте, что вместо клавиатуры приложение отображает красивый, реалистичный диск набора номера. Диск набора номера тщательно проработан, поэтому пользователи оценят его качество и мгновенно поймут, как его использовать. Набор номера реалистичен, и пользователи в восторге от жестов набора номера и сопутствующих звуков. Но для пользователей, которым часто требуется набирать номера телефона, которые не в списке контактов, начальное признание качества удобства использования вскоре сменится разочарованием, потому что с помощью роторного наборного диска набор менее эффективен, чем при использовании клавиатуры. В приложении, которое предназначено для совершения телефонных звонков, красивый UI обычно является препятствием.



С другой стороны, рассмотрим модель BubbleLevel приложения, которое отображает реалистичное исполнение строительного уровня. Люди знают, как использовать физический инструмент, значит они должны сразу знать, как использовать приложение. Приложение могло бы отобразить всю информацию без использования BubbleLevel, но это сделало бы приложение менее интуитивным и более сложным в использовании. В этом случае UI не только показывает людям, как использовать приложение, но и упрощает задачу.



Индивидуализация может, как улучшить, так и отвлечь внимание пользователя от задачи приложения, не забывайте эти руководящие принципы.

Всегда имейте основание для индивидуализации. В идеале, индивидуализация UI облегчает пользователям выполнение задачи и увеличивает их опыт. Вам требуется позволить задачам приложения управлять индивидуализацией ваших решений насколько это возможно. Например:

- Если ваше приложение позволяет продуктивно решать задачу, которая включает в себя манипуляции и много подробной информации, люди, скорее всего, занизят оценку вашего

приложения, отдавая предпочтения стандартному пользовательскому интерфейсу и эффективной навигации.

- Если ваше приложение помогает пользователям просматривать содержимое, то пользователи не оценят UI конкурирующее с ним.
- Если ваше приложение или игра обеспечивает захватывающий сюжетом опыт, люди рассчитывают погрузиться в уникальный мир, наполненный богатой, красивой графикой и инновационными взаимодействиями.

Избегайте, насколько это возможно, когнитивной нагрузки пользователя. Пользователи хорошо знакомы с внешним видом и поведением стандартных элементов UI, поэтому они не должны останавливаться и думать о том, как их использовать. Когда они сталкиваются с элементами, которые не выглядят или не ведут себя как стандартные, пользователи теряют преимущества своего предыдущего опыта. Пока ваши уникальные элементы не будут упрощать выполнение задачи, пользователям не будет нравиться изучение новых процедур, неиспользуемых в других приложениях.

Будьте внутренне последовательными. При увеличении индивидуализированных элементов UI требуется еще больше внимания обращать на внешний вид и поведение этих элементов, которые не должны быть противоречивыми в рамках вашего приложения. Если пользователи тратят время на изучение использования незнакомых элементов управления, созданных вами, они ожидают, что будут иметь возможность использовать это знание во всем вашем приложении.

Всегда считайтесь с контентом. Поскольку стандартные элементы хорошо знакомы, они не конкурируют с контентом за внимание пользователей. При индивидуализации UI позаботьтесь, чтобы он не заслонял необходимое пользователям содержание. Например, если ваше приложение позволяет пользователям просматривать видео, вы можете выбрать для дизайна пользовательских элементов индивидуализацию контроллеров воспроизведения. Но в любом случае, используете ли вы собственные или стандартные контроллеры управления воспроизведением, следует помнить, что они должны скрываться в начале просмотра видео и появляться при прикосновении.

Подумайте дважды, прежде чем изменять дизайн стандартных контроллеров. Если вы планируете индивидуализировать стандартный элемент управления, убедитесь, что измененный элемент не менее информативен, чем стандартный. Например, если вы создаете кнопку, которая не будет иметь округлый пространственный вид, ассоциируемый у пользователей с кнопками, пользователи не смогут понять, что ее можно нажимать. Или, если вы создаете переключатель, не показывающий противоположное положение, пользователи не смогут понять, что этот переключатель имеет 2 состояния.

Не забудьте тщательно протестировать индивидуализированные UI элементы. Во время тестирования, проконтролируйте, чтобы пользователи видели и могли предсказать, действия ваших элементов и могли легко взаимодействовать с ними. Если, например, вы создаете контроллер размером меньше, чем 44 x 44, люди столкнутся с проблемой нажатия. Если вы создаете окно, по-разному реагирующее на различные нажатия, обдумайте необходимость этого.

Prototype and Iterate (Прототип и Итерации)

Прежде чем инвестировать значительные ресурсы в разработку вашего проекта, хорошей идеей было бы создание прототипа для тестирования пользователями. Даже если у вас есть только несколько коллег для тестирования прототипов, вы сможете использовать их свежее видение функциональности вашего приложения и удобства использования.

На ранних этапах разработки проекта вы можете использовать бумажные прототипы или макеты для визуализации окон и контроллеров, а также карту приложения. Хотя вы можете получить полезную информацию от тестирования макетов, их неконструктивность может ввести в заблуждение тестеров. Это происходит, потому что пользователям трудно представить изменения опыта в работе приложения при заполнении макетов реальным содержанием.

Вы получите более ценные отзывы, если вы сможете собрать расширенный прототип, работающий на устройстве. Когда пользователи смогут взаимодействовать с прототипом на устройстве, они с большей вероятностью обнаружат места, где ваше приложение не отвечает их ожиданиям, или пользовательский опыт слишком сложен.

Самый простой способ создания надежного прототипа - использование одного из шаблонов Xcode с заполнением его соответствующим содержанием. (раскадровка файла охватывает весь пользовательский интерфейс вашего приложения, в том числе переходы между различными экранами.) Затем установите на устройство прототип, для того чтобы тестеры, по возможности, могли оценить реальный опыт

Вам не требуется предоставлять большое количество содержания или включать все элементы управления в прототип приложения, но вы должны обеспечить достаточный контекст, чтобы предложить реалистичный опыт. Стремитесь к балансу между типичной работой пользователей и более необычными крайними случаями. Например, если ваше приложение будет работать с длинными списками элементов, избегайте создания прототипа, который отображает только один или несколько элементов списка. Пока тестеры могут воспользоваться частью экрана для перехода к следующей логической цели или выполнению основной задачи, они в состоянии дать конструктивные отзывы.

Когда вы основываете свой прототип на шаблоне Xcode, вы получаете множество функциональных возможностей для свободной и довольно легкой интеграции дизайна и его корректировки в ответ на критику. В сжатые сроки выполнения работ, вы должны быть готовы протестировать несколько итераций разработки вашего прототипа, прежде чем утвердить свой проект и выделить ресурсы для его реализации. Чтобы начать изучение Xcode, см. "A Tour of Xcode".

Case Studies: Transitioning to iOS

Основы: Переход в iOS

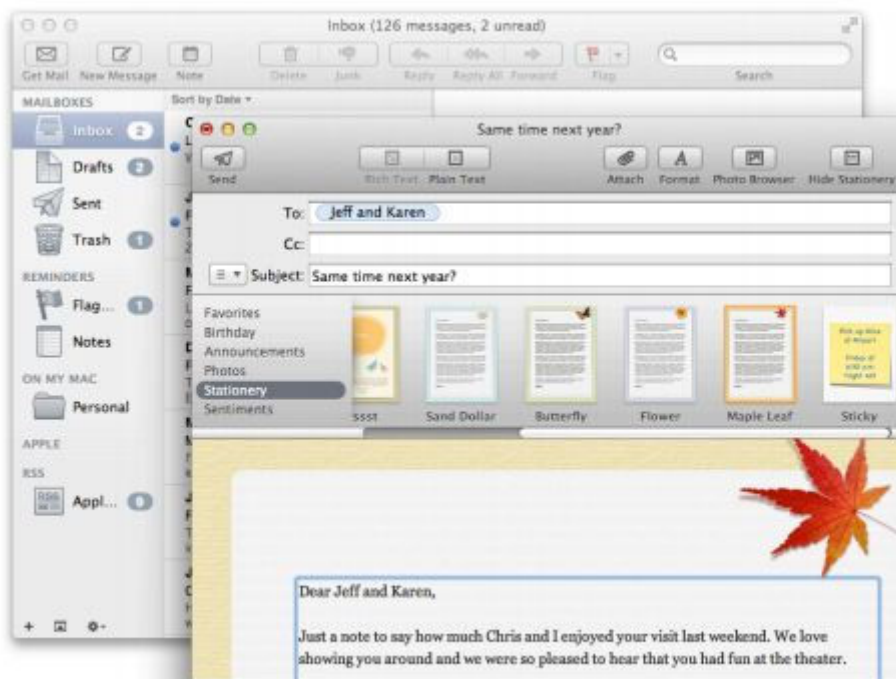
Вместо того чтобы создавать новые приложения, вы, возможно, планируете перенести существующие части программного обеспечения на платформу iOS. В некотором смысле реорганизация приложения или web-сайта для новой платформы может быть более сложной, чем разработка с нуля.

Приступая к такому проекту, рассмотрим, как пользователи используют iOS-устройства:

- Пользователи используют iOS-устройства совершенно иначе, нежели они используют настольные и портативные компьютеры, и они имеют разные ожидания для пользовательского опыта. Перенос программного обеспечения с компьютера на iOS-устройства редко является простым.
- Пользователи часто используют iOS-устройства и в обстановке, заставляющей отвлекаться. Часть вашей работы заключается в создании ответа на вынужденный опыт, который привлекает пользователей к быстрому и легкому получению содержания.
- Помните правило 80-20: В общем, самый большой процент пользователей (по крайней мере 80 процентов) используют очень ограниченное число функций в приложении, и только небольшой процент (не более 20 процентов) пользуются всеми функциями. В iOS-приложениях редко возникает необходимость обеспечивать все функции управления, которые могут понадобиться пользователям.

From Mail on the Desktop to Mail on iPhone (От Почты На Рабочем Столе К Почте На iPhone)

Почта является одним из наиболее посещаемых, хорошо используемых, и высоко оцененных приложений в Mac OS X. Это также очень мощная программа, которая позволяет пользователям создавать, получать, расставлять приоритеты, хранить электронную почту, отслеживать действия предметов и событий, а также создавать заметки и приглашения. Desktop-почта предлагает эту мощную функциональность в нескольких окнах.



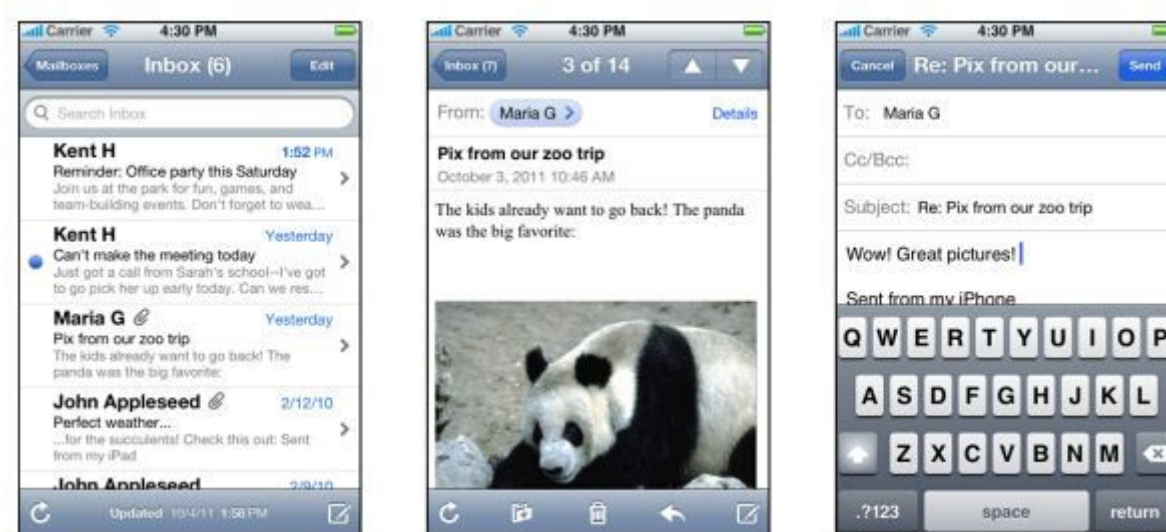
Почта на iPhone фокусируется на основных функциональных возможностях desktop-почты, чтобы помочь людям получать, создавать, отправлять и организовывать свои сообщения. Почта на iPhone обеспечивает эту функциональность в сжатом UI, специально предназначенном для мобильного взаимодействия, который включает в себя:

- Четкий вид, отображаемый по центру и на переднем плане;
- Различные окна, направленные на облегчение выполнения различных задач;
- Интуитивно понятная структура информации, масштабируемая без каких-либо усилий;
- Мощные инструменты редактирования и организации, доступные при необходимости;
- Едва заметные, но выразительные анимации, сопровождающие действия и обеспечивающие обратную связь.

Важно понимать, что почта на iPhone не лучше, чем Desktop-почта, а, скорее, это почта, разработанная для мобильных пользователей. Концентрируясь на подмножестве desktop-функций и представлении их в привлекательном и простом UI, почта на iPhone предоставляет пользователям ядро почтового функционала, пока они мобильны.

Для адаптации опыта использования Почты, почта на iPhone имеет инновационный UI разрабатываемый в нескольких ключевых направлениях.

Особые, узкоспециализированные экраны. Каждый экран отображает один аспект почтового опыта (mail experience): список учетных записей, список почтового ящика, список сообщений, отображение сообщения и структурное отображение. В пределах экрана пользователи осуществляют прокрутку для просмотра всего содержимого.



Легкая, предсказуемая навигация. Делая одно нажатие на экран, пользователи углубляются от общего (список аккаунтов) к частному (сообщение). Каждый экран отображает название, которое показывает пользователям, где они находятся, и кнопку назад, что делает легким отслеживания их шагов.

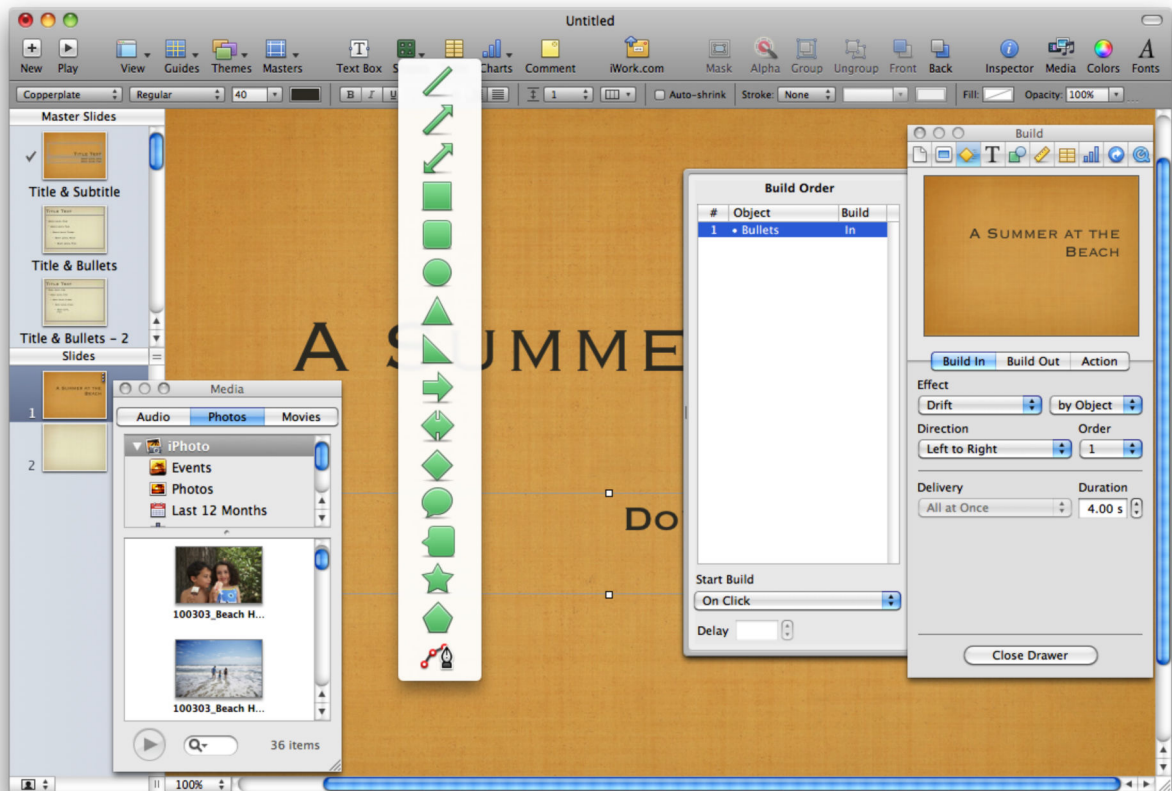
Простые нажимающиеся контроллеры, доступные, когда требуется. Поскольку написание сообщений и проверка новых сообщений электронной почты являются первичными действиями людей, возможно использующимися в любом контексте, Mail на iPhone делает их доступными в нескольких экранах. Когда люди просматривают сообщение, такие функции, как “ответить”, “переместить”, и “удалить” доступны, поскольку они применимы к сообщениям.



Различные виды обратной связи для различных задач. Когда люди удаляют сообщение, происходит анимация перемещения в значок корзины. Когда люди посылают сообщение, они могут видеть его прогресс, а когда отправление завершено они могут услышать характерный звук. Глядя на едва заметный текст в панели инструментов списка сообщений, пользователи могут сразу увидеть, когда их почтовый ящик последний раз обновлялся.

From Keynote on the Desktop to Keynote on iPad (От Презентации на Рабочем Столе к Презентации на iPad)

Desktop-кейноут представляет собой мощное, гибкое приложение для создания слайдовых презентаций мирового класса. Пользователям нравится, как кейноут сочетает в себе простоту использования с мелкой структурой контроллеров над множеством точных деталей, таких как анимация и атрибуты текста.



Кейноут на iPad отражает суть desktop-кейноута и делает ее родной на iPad путем создания пользовательского восприятия:

- Основное внимание уделяется пользовательскому содержанию
- Снижение сложности без потери потенциала
- Обеспечение ярлыками, которые помогают и восхищают
- Адаптация знакомых признаков desktop-восприятия
- Обеспечение обратной связи, а также связь через яркие анимации

Пользователям кейноут позволяет мгновенно понять, как использовать приложение на iPad, поскольку обеспечивает ожидаемую функциональность с помощью собственных парадигм iPad. Новым пользователям легко узнать, как использовать кейноут на iPad, поскольку они могут напрямую управлять его содержанием простыми, естественными способами.

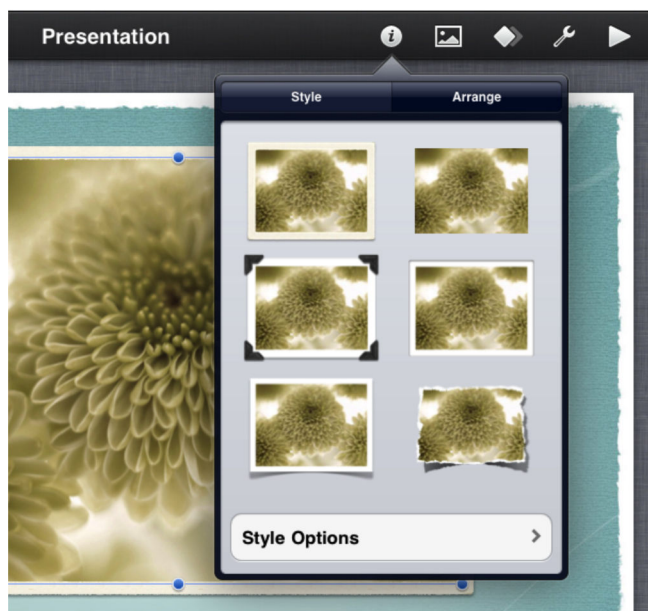
Преобразование кейноута от desktop-систем до iPad основано на множестве модификаций и модернизаций, которые варьируются от едва заметных до основательных. Вот некоторые из наиболее заметных адаптаций:

Рационализированные панели инструментов. Всего семь элементов в панели инструментов, но они дают пользователям постоянный доступ ко всем функциям и средствам, необходимым для создания содержания.

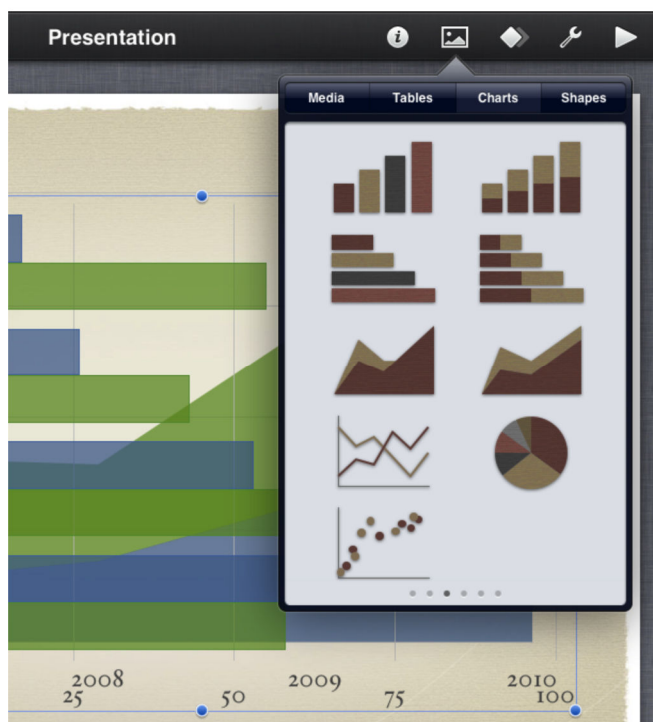


Упрощенный приоритетный контроллер, реагирующий на то, на что сфокусирован пользователь. Кейноут-контроллер на iPad автоматически содержит инструменты и атрибуты,

необходимые пользователям для изменения выбранного объекта. Часто люди могут сделать все изменения, которые им необходимы, в первом окне контроллера. Если они должны изменяться реже изменения атрибутов, они могут перейти к другим окнам контроллера.



Много готовых коллекций стиля. Люди легко могут изменять внешний вид объектов, таких как диаграммы и таблицы, за счет использования готовых стилей. В дополнение к цветовой схеме каждая коллекция включает в себя prestyled-атрибуты, такие как заголовки таблицы и оси деления маркировок, разработанные для координирования своих действий с общей темой.

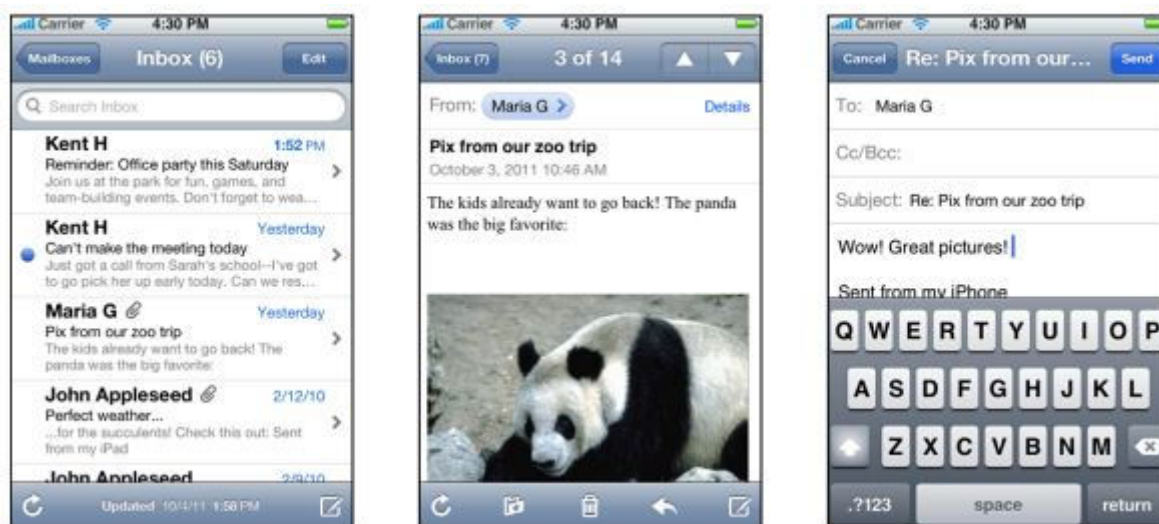


Прямая манипуляция содержанием, обогащенного выразительной анимацией. В iPad кейноуте пользователь перетаскивает слайд на новую позицию, вращая объект, чтобы повернуть его, и нажимает на изображение, чтобы выбрать его. Впечатление от

непосредственной работы увеличивается за счет реагирования на основную анимацию, выполняемую на iPad. Например, слайд немного пульсирует, когда пользователи перемещают его, и когда они перетаскивают его на новое место, окружающие слайды колеблются, освобождая для него место.

From Mail on iPhone to Mail on iPad (От Почты На iPhone К Почте На iPad)

Почта является одним из ведущих встроенных приложений на iPhone. Люди ценят, что оно понятное, рационализированное, и как оно организует большие объемы информации в серии простых в использовании экранов.



Почта на iPad предоставляет те же самые функции. В тоже время она предоставляет больше экранного пространства для пользовательских сообщений, реалистичные мощные средства организации и редактирования, которые всегда доступны, но ненавязчивы. Визуальная стабильность и лаконичность экрана обеспечивают то, что требуется пользователям в данном месте, с минимальными изменениями контекста.

Различия между почтой на iPhone и почтой на iPad в том, что используется разное пользовательское восприятие каждого устройства.

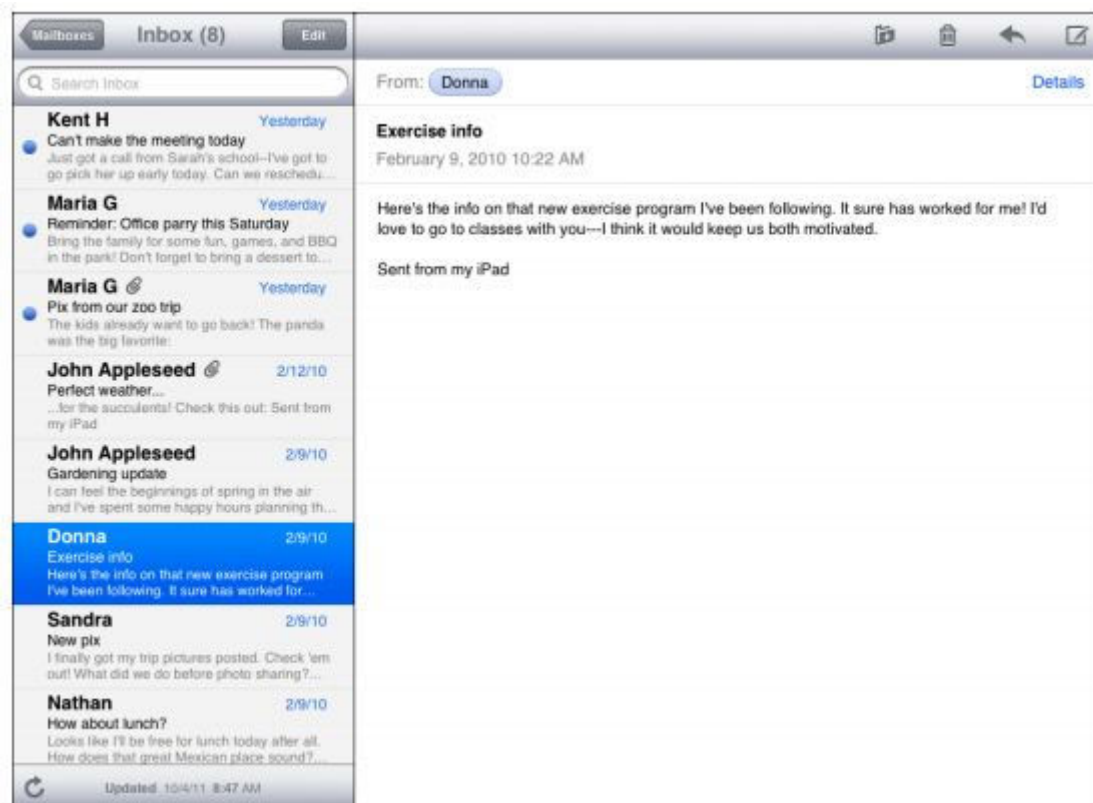
- Почта на iPhone предназначена для мобильных пользователей, позволяя обрабатывать свою электронную почту, пока они стоят в очереди или отправились на встречу.
- Почта на iPad достаточно эффективна для людей в использовании на ходу, но ее богатый опыт также подразумевает и углубленное использование.

Хотя почта на iPad привязывает пользовательское восприятие к устройству, она не меняет базовую пользовательскую функциональность, местоположение или действия отдельных функций. Пользователи iPhone-почты легко распознают элементы панели инструментов в структуре почтовых ящиков в iPad-почте и сразу знают, как их использовать, поскольку они практически идентичны.

Для улучшения мобильного восприятия почты приложение Mail на iPad улучшается по сравнению с приложением Mail на iPhone в ряде некоторых моментов:

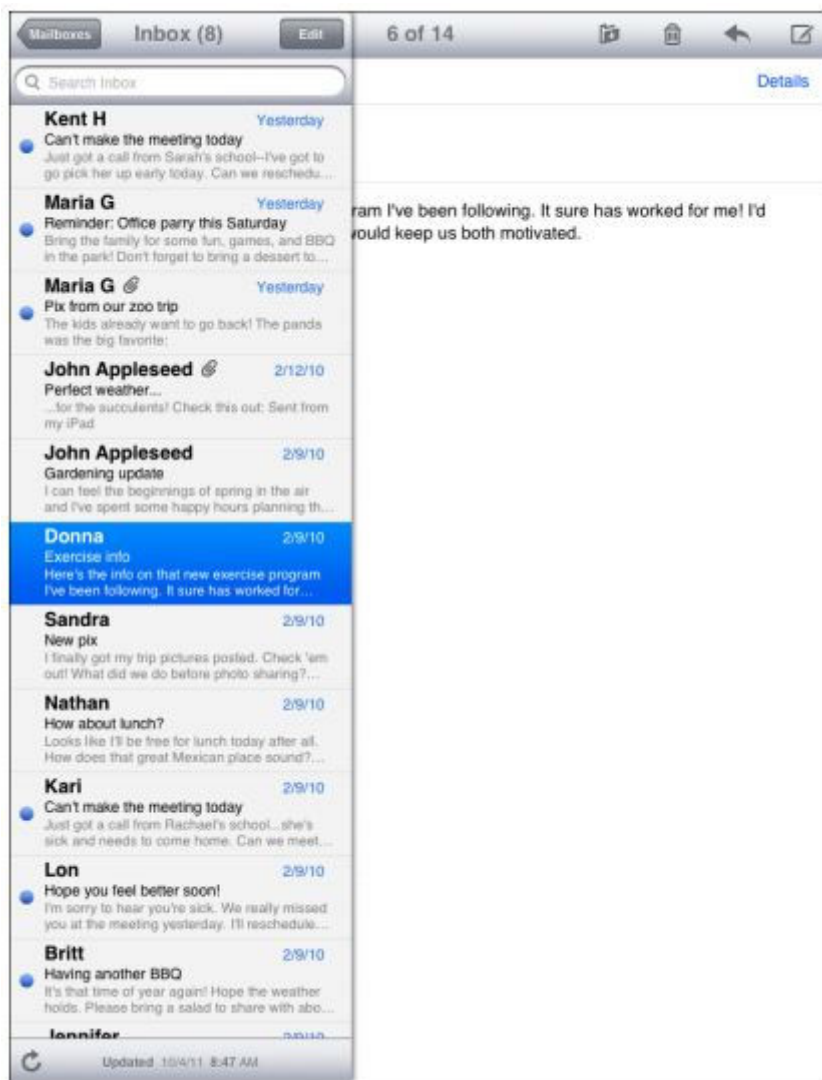
Расширенная поддержка ориентаций устройства. Пользователи могут использовать Mail на

iPad в любой из четырех ориентаций (горизонтальная ориентация показана здесь). Хотя горизонтальная ориентация несколько отличается от вертикальной, фокусировка остается на содержании и функциональности.



Сосредоточение внимания на содержимом сообщения. Mail на iPad резервирует большую часть экрана для текущего сообщения в любой ориентации (горизонтальная ориентация показана на следующем рисунке).

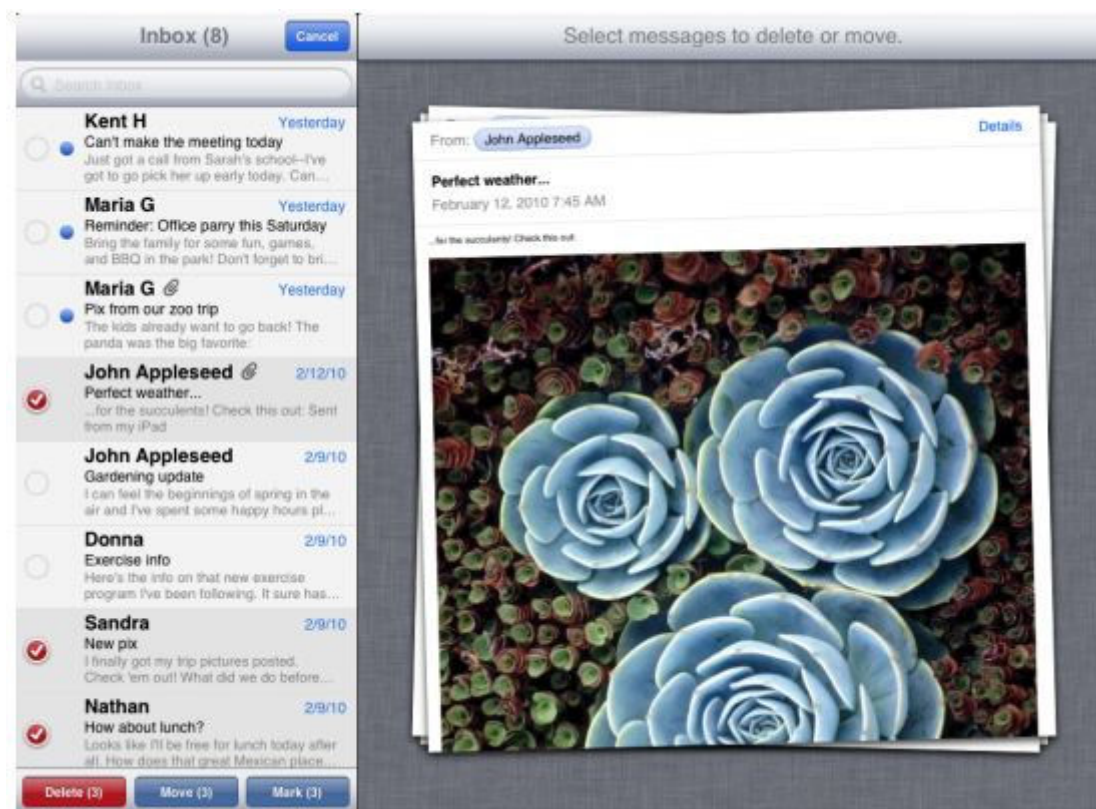
Это включает в себя движущиеся панели инструментов в верхней части сообщения, с целью увеличения вертикального пространства, доступного для содержимого сообщения. В дополнительном пространстве пользователи могут читать длинные сообщения с меньшим количеством прокрутки. А когда пользователи хотят просмотреть список сообщений в горизонтальной ориентации, они все еще могут видеть большую часть текущего сообщения.



Достоинства иерархии. Почта на iPad эффективно структурирует почтовую иерархию (то есть “учетная запись > почтовый ящик > Список сообщений > сообщение”), ограничивая все уровни выше самого сообщения в отдельных UI-элементах. В горизонтальной ориентации эти уровни постоянно отображаются в левой части отдельного окна (split view); в вертикальной - всплывающее окно, которое всплывает и отображается поверх главного экрана.

Значительное снижение полноэкранных переходов. Поскольку большая часть иерархии доступна в отдельном экранном элементе, пользователи могут получить доступ к большинству из того, в чем они нуждаются на отдельном экране. Когда пользователи перемещаются вниз по иерархии, это “учетная запись > почтовый ящик > окно списка сообщений”, все переходы осуществляются не на весь экран.

Реалистичные сообщения. Когда пользователи помечают письмо для удаления, оно скользит по окну сообщений аналогично физическому листу бумаги. Когда они выбирают дополнительные сообщения для удаления, эти сообщения формируются в реалистичную стопку бумаг со слегка помятыми краями.



From a Desktop Browser to Safari on iOS (От Desktop-Браузера К Safari На iOS)

Safari на iOS обеспечивает ощущение web-просмотра при использовании на iOS-устройствах. Люди ценят четкость текста и изображений с возможностью регулирования их отображения вращением устройства, или пощипыванием, или нажатием на экран.

Сайты, основанные на стандартах web, хорошо отображаются на iOS-устройствах. В частности, web-сайты, обнаруживаемые устройством и не использующие плагинов, отлично смотрятся как на iPhone, так и на iPad с небольшими, если таковые имеются, изменениями.

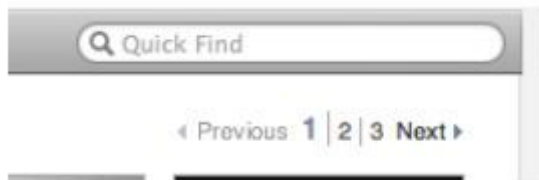
Кроме того, самые успешные web-сайты обычно:

- Настраивают окно просмотра соответственно для устройства, если необходимо, чтобы ширина страницы соответствовала ширине устройства.
- Избегают фиксированного позиционирования CSS так, чтобы содержание не выходило за экран, когда пользователи масштабируют или перетаскивают страницу.
- Включают сенсорный UI, который не использует указателей взаимодействия

Иногда другие модификации могут быть целесообразными. Например, web-приложения всегда устанавливаются по ширине окна должным образом и часто скрывают UI Safari на iOS. Чтобы узнать больше о том, как сделать эти изменения, читайте "Configuring the Viewport" и "Configuring Web Applications" в "Safari Web Content Guide".

Web-сайты адаптируют web-восприятие от Safari на рабочем столе компьютера в iOS, также и в следующем:

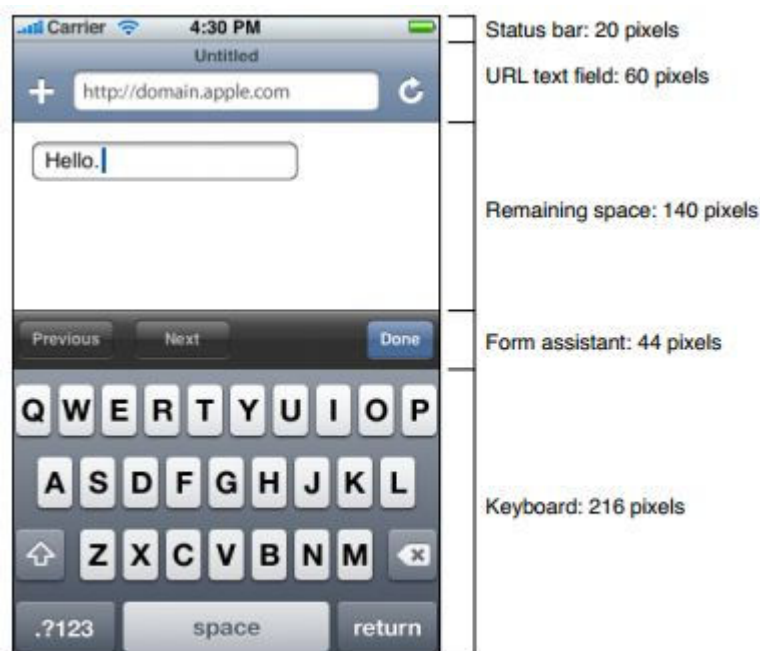
Использование индивидуального CSS для адаптации UI. Различные UI-элементы могут применяться в различных средах. Например, сайт Apple включает в себя страницу, которая отображает фильмы, доступные для просмотра. При просмотре в desktop Safari пользователи могут нажать на номера предыдущих и следующих элементов управления, чтобы увидеть дополнительные анонсы:



При просмотре на iPhone контроллеры «следующий», «предыдущий» и «цифры», являются заметными, простыми в использовании кнопками с символами, которые повторяют стиль встроенных элементов управления (таких, как индикатор подробного раскрытия информации и индикатор страниц):



Размещение клавиатуры в Safari на iOS. Когда видны клавиатура и form assistant, то Safari на iPhone отображает web-страницы в области, ниже текстового поля URL и выше клавиатуры и form assistant.

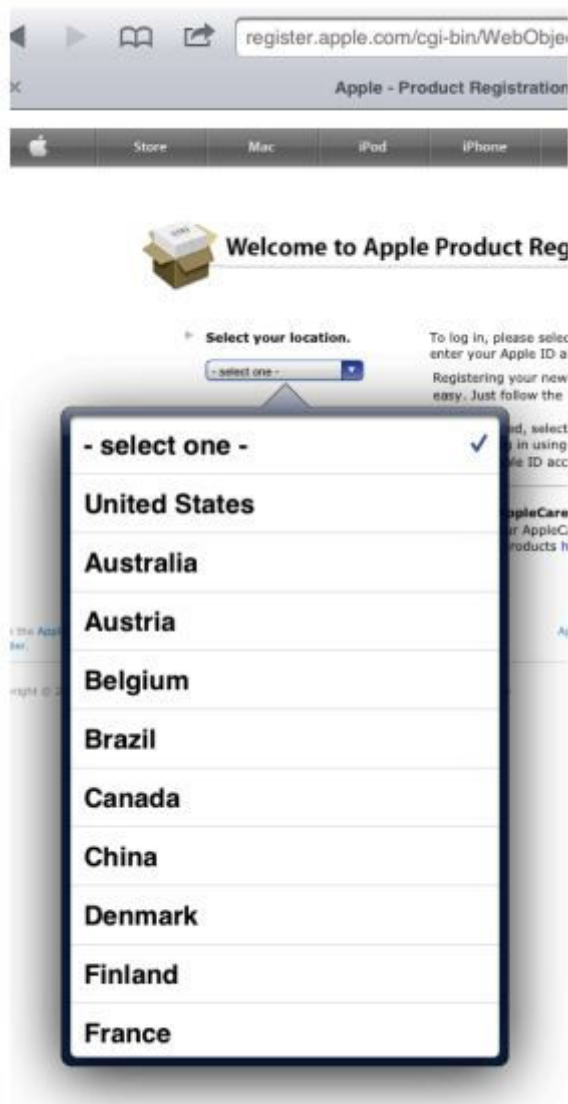


Когда клавиатура и form assistant не отображаются, то в наличии есть дополнительные 216 пикселей по вертикали (в вертикальной ориентации) для отображения web-страницы. В горизонтальной ориентации эти два значения отличаются: высота клавиатуры составляет 162 пикселя, а высота form assistant составляет 32 пикселя.

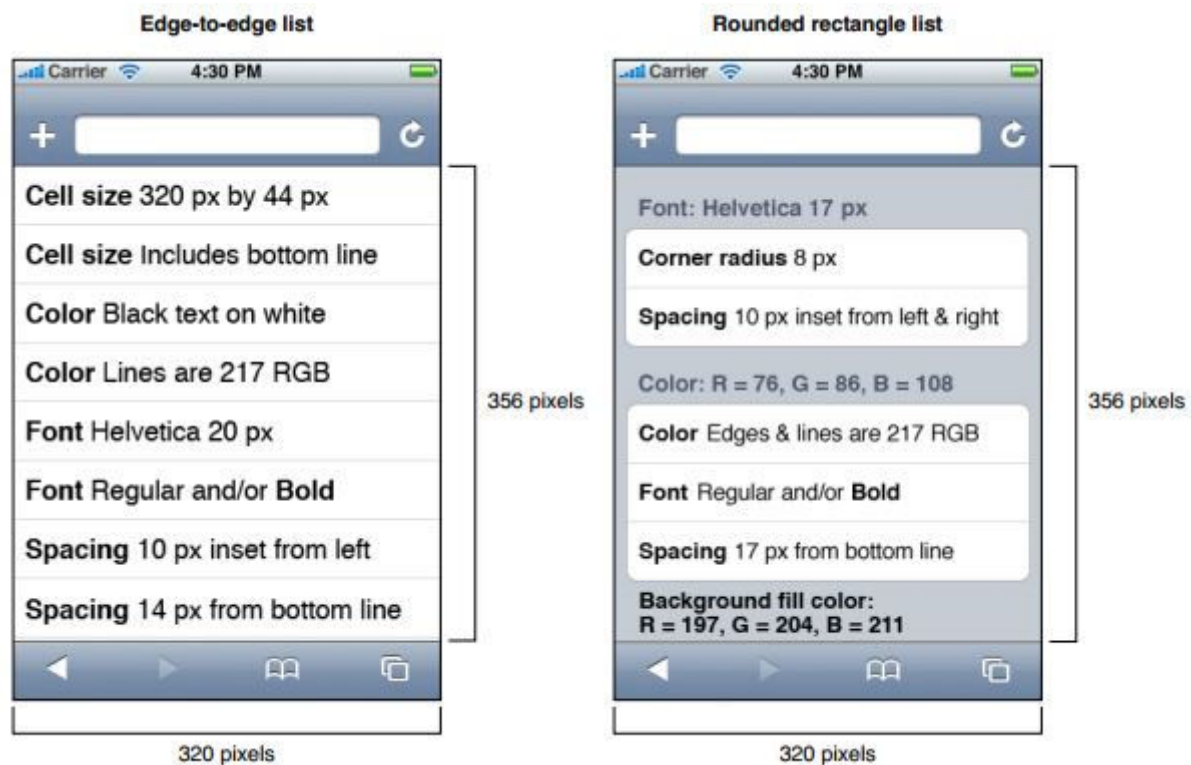
Размещение всплывающего меню управления в Safari на iOS. В браузере Safari на desktop рор-уп меню, содержащее большое количество элементов отображается, как в Mac OS X приложениях, то есть открывается меню, и для того, чтобы отобразить все элементы, оно расширяется за границы окна, если это необходимо. В браузере Safari на iOS всплывающее меню отображается при помощи родного элемента, обеспечивающего гораздо лучшее пользовательское восприятие. Например, на iPhone всплывающее меню появляется в **пикере** списка вариантов, из которых пользователь может выбрать, как показано ниже.



На iPad всплывающее меню отображается во всплывающем окне, как показано ниже.



Использование списков для отображения данных в iPhone web-приложениях. iOS пользователи привыкли к спискам, присущим приложениям, поэтому, когда они видят списки в web-приложениях, они, вероятно, думают, что это web-контент в приложении. На iPhone списки отображаются от края до края или внутри прямоугольников со скругленными углами. Каждый стиль определяется его собственными показателями.



User Experience Guidelines

Руководство по Пользовательскому Впечатлению

Пользовательский опыт взаимодействия iOS-устройств, вращается вокруг прямого взаимодействия пользователя и контента. Руководство в этой главе применимо ко всем устройствам, основанным на iOS.

Focus on the Primary Task

(Фокусируйтесь на Основной Задаче)

Когда приложение iOS устанавливает и поддерживает фокус на своей основной задаче, это устраивает пользователя и доставляет удовольствие в использовании. Определение документации вашего приложения поможет вам сфокусировать приложение на его основной задаче (чтобы узнать больше, смотрите “Create an Application Definition Statement” (стр. 19)) Чтобы поддерживать этот фокус, вам необходимо определить, что является наиболее важным в каждом контексте или экране.

Анализируйте, что необходимо на каждом экране. Как только вы решили, что отображать на каждом экране, спросите себя - критична ли данная информация или функциональность и необходима ли она пользователям прямо сейчас? Если вы ответите "нет", то решите, критична ли данная информация или функционал, или она совсем не важна.

Например, приложение Calendar для iPhone сфокусировано на днях и на событиях, которые в это время происходят. Пользователи могут использовать четко названные кнопки, чтобы выделить текущий день, выбрать опцию просмотра и добавить события.



Elevate the Content People Care About (Выделяйте Контент, Который Нужен Пользователям)

В игре пользователи заботятся об опыте (переживаниях); они не ожидают управления, удаления или создания контента. Если вы разрабатываете игру, вы выделяете контент по важности опыта при помощи истории, прекрасной графики и геймплея.

Если вы не разрабатываете игру, вы можете помочь пользователям сфокусироваться на контенте, создавая дизайн UI как искусное обрамление информации, в которой они заинтересованы. Вот несколько способов как этого добиться:

Минимизируйте количество и выделение контроллеров, чтобы уменьшить их "вес" в UI.

С фотографиями все точно так же располагайте несколько ненавязчивых контроллеров на прозрачных панелях.

Обдумайте небольшую индивидуализацию контроллеров, чтобы они интегрировались в графический стиль вашего приложения. В этом случае контроллеры ясны и понятны, без вычурности.

Подумайте о скрытии контроллеров после того, как пользователи перестали с ними взаимодействовать какое-то время, и повторном отображении, когда пользователи снова нажимают на экран. Иногда вы возможно захотите скрыть также и остальной UI приложения. Это особенно необходимо в приложениях, которые предоставляют глубокие впечатления, потому что они предоставляют даже больше, чем просто размер экрана, чтобы донести содержание до пользователей. Например, приложение Photos скрывает контроллеры и панели после того, как некоторое время не было взаимодействия, что в свою очередь направляет пользователей на погружение в содержание. Когда пользователи хотят выполнить какую-либо задачу с их фотографиями, одно нажатие где-либо на экране вновь отображает контроллеры.

Think Top Down (Думайте "Сверху вниз")

Верх экрана - это наиболее видимая пользователям часть, потому что они тяготеют ко взаимодействию с устройством, держа устройство следующими способами:

- В своей неосновной руке (или лежа на поверхности) и производя жесты пальцем основной руки.
- В одной руке и производя жесты большим пальцем той же самой руки.
- Между двумя руками, производя жесты большими пальцами обеих рук.

Расположите наиболее часто используемую (обычно наивысший уровень) информацию сверху, где она наиболее лучше видна и наиболее доступна. Так как пользователь просматривает экран сверху вниз, отображаемая информация должна прогрессировать от общему к частному и от высокого уровня к низкому.

Например, в игре наиболее важное действие может располагаться в верхней половине экрана. Это освобождает нижнюю половину экрана для дополнительной информации, а также для контроллеров, на которые пользователи могут нажать без перекрывания общего окна приложения.

Give People a Logical Path to Follow (Дайте Пользователям Логический Путь для Следования)

Сделайте путь через информацию, которую вы предоставляете, логическим и предсказуемым для пользователей. В дополнение, убедитесь, что вы предоставляете маркеры, такие как кнопка "назад", которые пользователь может использовать, чтобы понять, где он находится и как отследить свои шаги.

В большинстве случаев предоставляйте пользователям только один путь к экрану. Если экран должен быть доступен в различных обстоятельствах, подумайте об использовании модального окна, которое может появляться в различных контекстах.

Make Usage Easy and Obvious (Сделайте Использование Простым и Очевидным)

Попытайтесь сделать ваше приложение действительно понятным пользователям, потому что вы не можете быть уверены, что у них есть время (или они могут потратить на это свое внимание), чтобы понять, как оно работает.

Сделайте так, чтобы основная функция вашего приложения была сразу же очевидна. Вы можете добиться этого следующим:

- Уменьшение количества контроллеров, из которых пользователи должны выбирать.
- Используйте стандартные контроллеры и жесты должным образом, в соответствии с ожидаемым поведением.
- Четко называйте контроллеры, чтобы пользователи смогли точно понять, что они делают

Будьте последовательны в использовании парадигм встроенных приложений. Пользователи понимают, как проводить навигацию по иерархии экранов, редактировать содержание списка, и переключаться между режимами приложения, используя панель Ярлыков (TabBar). Сделайте так, чтобы пользователям было легко использовать ваше приложение, путем усиления их впечатления. Например, пользователи понимают, что синяя кнопка на панели инструментов представляет собой действие, поскольку они понимают это в текущем контексте. Если ваше приложение отображает больше одной синей кнопки в одном контексте, пользователи будут иметь выбор, на какую из кнопок нажать.

Во встроенной функции Stopwatch (часть iPhone-приложения Clock) пользователи могут легко увидеть, какая кнопка останавливает и запускает снова секундомер, и какая кнопка "ловит" время круга.



Use User-Centric Terminology (Используйте Направленную на Пользователя Терминологию)

Во всех ваших взаимодействиях с пользователями, основанных на тексте, используйте терминологию, в которой вы уверены, что она будет понятна пользователю. В частности, избегайте технического жаргона в пользовательском интерфейсе. Используйте то, что вы знаете о своих пользователях, чтобы определить, являются ли слова и фразы, которые вы планируете использовать, понятными.

Экран Wi-Fi Networks Settings (Настройки Сетей Wi-Fi) использует простой язык для объяснения - как iOS отвечает на пользовательские предпочтения.



Позаботьтесь о том, чтобы быть точным при описании даты. Для этого часто целесообразно использовать дружеские отношения, такие как “сегодня” и “завтра” при отображении актуальной информации в пользовательском интерфейсе, но это может ввести в заблуждение, если вы не учитываете текущую локацию пользователя. Например, рассмотрим момент, который начинается незадолго до полуночи. Для пользователей в одинаковом часовом поясе, события начинаются сегодня, а для пользователей, находящихся в другом часовом поясе (более раннем), то же самое событие, возможно, началось вчера.

Minimize the Effort Required for User Input (Минимизируйте Усилия, Необходимые Пользователю для Ввода Данных)

Ввод информации занимает время и внимание, используют ли пользователи нажатие на контроллеры или используют клавиатуру. Если в вашем приложении пользователю необходимо ввести большое количество данных, прежде чем произойдет что-то полезное, то данный ввод информации замедляет пользователя и может служить препятствием для использования вашего приложения.

Сбалансируйте каждый запрос между вводом данных от пользователей и тем, что вы предлагаете пользователям взамен. Другими словами, пытайтесь предоставить как можно больше информации или функционала для каждой части информации, которую пользователи передают вам. Таким образом, пользователи чувствуют, что происходит прогресс и не происходит задержки в работе приложения.

Сделайте ввод выбора пользователя как можно проще. Например, вы можете использовать окно таблицы или пикер вместо текстового поля, потому что пользователям обычно проще выбрать какой-то пункт из списка, чем напечатать слова. (Чтобы узнать больше об этих элементах пользовательского интерфейса, см. раздел “Table View” (page 119) и “Picker” (page 146).)

Получайте информацию из iOS, когда это необходимо. Пользователи сохраняют очень много информации на своих устройствах. Когда это имеет смысл, не заставляйте пользователей предоставлять вам информацию, которую вы можете легко найти сами, такую как контакты или календарь.

Downplay File-Handling Operations (Уменьшайте Файлозагрузочных Операций)

Хотя iOS приложения и позволяют пользователям создавать и манипулировать файлами, это не означает, что пользователи должны заботиться о файловой системе на устройствах, основанных на iOS.

Не существует iOS аналогов программы Mac OS X Finder и пользователи не должны взаимодействовать с файлами так, как они делают это на компьютере. В частности, пользователи не должны быть поставлены в такую ситуацию, которая будет требовать от них обдумывания мета-данных файла или места расположения, такую как:

- Открытие или сохранение диалога, который открывает файловую иерархию
- Раскрытие информации о разрешенном статусе файлов

По мере возможности позволяйте людям управлять документами без открытия iTunes на их компьютере. Рассмотрите возможность использования iCloud-хранение, чтобы пользователи получили доступ к своему содержимому на всех своих устройствах. Чтобы получить несколько советов о том, как обеспечить больший iCloud-опыт в вашем приложении, см. “iCloud Storage” (стр. 68).

Если ваше приложение позволяет пользователям создавать или редактировать документы, то необходимо предоставить какой-то вид пикера документа, который позволяет им открывать существующий документ или создавать новый. В идеале данный пикер документа:

- **Выполнен в высоком качестве графики.** Пользователи должны иметь возможность легко идентифицировать документ, увидев визуальную презентацию документа на экране.
 - **Позволяет пользователям в нескольких доступных жестах сделать то, что они хотят.** Например, пользователи могут произвести горизонтальный скролл через все существующие документы и открыть необходимый одним нажатием.
 - **Включает функцию нового документа.** Вместо того, чтобы заставлять пользователей идти куда-то еще для создания нового документа, пикер документа может позволить им нажать на картинку заполнения для создания нового документа.
- Вы так же можете использовать функцию Быстрого Просмотра (Quick Look Preview), чтобы позволить пользователям предварительно просмотреть документ внутри вашего приложения, даже если приложение не может его открыть. Чтобы узнать больше о предоставлении данной функции в вашем приложении, смотрите “Quick Look Document Preview” (стр. 83).

Enable Collaboration and Connectedness (Предоставление Сотрудничества и Связей)

iOS-устройства являются личными устройствами, но они так же могут предоставлять возможность взаимодействия с другими пользователями. Улучшите ваше приложение, помогая пользователям во взаимодействии и связями с другими пользователями.

Когда это необходимо, сделайте так, чтобы было очень просто пользователям взаимодействовать с другими людьми и чем-то делиться, например – их местом расположения, мнениями и заработанными в игре очками. Обычно пользователи ожидают возможности делиться (share) информацией - это очень важно для них.

Большинство приложений могут добавлять значения, позволяя пользователям переходить за пределы приложения и делиться данными с другими инструментами, которые они используют. Например, iOS приложение может действовать как мобильное дополнение к компьютерному приложению.

Или, например, iPad-приложение может позволить своим пользователям взаимодействовать с пользователями iPhone-версии приложения.

Если ваше приложение позволяет пользователям получить доступ к своему аккаунту Twitter, не забудьте использовать Twitter API, чтобы не заставлять пользователей многократно осуществлять вход на одном и том же устройстве.

На iPad обдумайте пути, которые позволят нескольким пользователям использовать приложение на одном устройстве. Например, два человека могут иметь возможность играть в игру на противоположных сторонах одного экрана. Или приложение-музыкальная группа может позволить различным людям играть на различных инструментах вместе на одном устройстве.

De-emphasize Settings (Не-выделение Настроек)

По возможности избегайте включения настроек в ваше приложение. Настройки включают в себя предпочтительные поведения приложений и информацию, которую пользователи редко хотят изменить. Пользователи не могут открыть приложение Настройки без выхода из вашего приложения, и вы не должны поощрять это действие.

Когда вы создаете дизайн вашего приложения так, чтобы оно функционировало в том виде, который наиболее ожидают пользователи, то вы уменьшаете необходимость в настройках. Если вам необходима информация о пользователе, запросите систему об этой информации, вместо того, чтобы просить пользователя предоставить ее. Если вы решите предоставить настройки в iOS-приложении, см. “The Settings Bundle in iOS Application Programming Guide” чтобы узнать, как поддержать их в коде.

Позвольте пользователям установить то поведение, которое они хотят, используя опции конфигурации в вашем приложении. Опции конфигурации позволят вашему приложению динамически реагировать на изменения, потому что пользователи не хотят покидать ваше приложение для их установок.

Предложите установить опции конфигурации на основном пользовательском интерфейсе (в iPhone приложениях) на обратной стороне окна. Чтобы определить какое их месторасположение имеет смысл, определите – предоставляют ли опции основную задачу и как часто пользователи могут захотеть их использовать.

- На основном UI установите опции, которые предоставляют основной функционал, или те, которые пользователи захотят часто изменять.

Например, iPad Calendar позволяет пользователям просматривать свои расписания за день, неделю или месяц. Эти опции конфигурации предоставлены на основном UI, потому что просмотр различных перспектив календаря – это основная функция приложения и пользователи хотят часто изменять ее.

Приложения, которые предоставляют глубокие впечатления, такие как игры, очень часто предоставляют опции конфигурации внутри приложения, потому что пользователи стремятся изменять данные аспекты своего впечатления очень часто.

- В iPhone приложениях, те опции, которые пользователи не так часто изменяют, вы можете расположить позади окна. Например, основная функция приложения Погода – это отображение текущего состояния погоды в городе и 6-и-дневный прогноз погоды. Это очень важно, чтобы пользователи имели выбор, в каком параметре отображать температуру – в Цельсиях или в Фаренгейтах, хотя пользователи обычно не так часто изменяют эту опцию. Тем самым имеет смысл разместить опцию смены шкалы температуры позади окна Погоды, где она постоянно доступна, но не навязчива.



Brand Appropriately (Уместное Использование Бренда)

Включайте цвета бренда или изображения в утонченном, ненавязчивом виде. Брендинг наиболее эффективен, когда он еле уловим. Пользователи используют ваше приложение, чтобы привести свои дела в порядок или чтобы развлечься; они не хотят чувствовать себя так, словно их заставили смотреть рекламу. Для обеспечения наилучшего пользовательского впечатления, вам необходимо слегка напомнить пользователям о вашей идентификации.

Избегайте захвата того пространства, которое необходимо пользователям. Например, отображение вторичной постоянной панели сверху экрана, в котором нет ничего, кроме отображения рекламы бренда, приводит к меньшему количеству пространства для контента. Обдумайте другие, менее навязчивые способы отображения общего бренда, такие как легкая индивидуализация фона экрана.

Исключением данного руководства является иконка приложения, которая должна быть полностью сфокусирована на вашем бренде. Так как пользователи достаточно часто видят вашу иконку, очень важно потратить немного больше времени, балансируя между привлекательностью и четким отображением бренда в иконке.

Make Search Quick and Rewarding (Сделайте функцию Поиска Быстрой и Удобной)

В приложениях, которые содержат или отображают множество данных, поиск может быть основной функцией. Если вам необходимо предоставить поиск в вашем приложении, следуйте данному руководству, чтобы убедиться, что он будет предоставлен наилучшим образом.

Встройте каталог ваших данных, чтобы приложение всегда было готово к поиску. Не ждите момента, что пользователь инициирует на это функцию поиска, потому что вам не нужно создать негативное представление о функции поиска в вашем приложении.

Используйте локальные данные Live-filter, чтобы приложение могло быстрее отображать результаты. Самое лучшее – когда вы можете начать фильтрацию, как только пользователи начинают набирать текст, и предоставлять результаты в тот момент, когда они продолжают печатать. Хотя живая фильтрация данных обычно приводит к отличному пользовательскому впечатлению, она не всегда является практичной. Когда живую фильтрацию нельзя применить, вы можете начать процесс поиска после того, как пользователь нажимает на кнопку Поиск на клавиатуре. В таком случае, убедитесь, что представляете отчет о прогрессе поиска, чтобы пользователь знал, что процесс не «затормозился».

Когда это возможно также используйте удаленную фильтрацию данных, когда пользователь набирает текст. Хотя фильтрация пользовательского набора текста приводит к лучшему пользовательскому впечатлению, будьте уверены, что вы можете проинформировать их и дать им возможность отказаться от этой опции, если время ответа, возможно, задержит результаты более чем на одну или две секунды.

Отображайте панель поиска над списком или заголовком списка. Пользователи ожидают найти панель поиска в этой позиции, потому что они уже привыкли к панели поиска в приложении Contacts и других приложениях. Расположение панели поиска в этом месте означает, что она не встает на пути пользователя, когда они прокручивают список или используют оглавление.

Если поиск является основной функцией вашего приложения, то вы, возможно, захотите вынести эту функцию в отдельный режим. В iTunes, например, нахождение и приобретение музыки и подкастов – это то, на чем концентрируется приложение. Пользователи хотят искать их любимые песни, исполнителей или подкасты, вне зависимости от того режима, в котором они находятся в текущий момент, и таким образом, имеет смысл предложить посвященный этому ярлык поиска, который всегда доступен.

Если это необходимо, отобразите заполнение контента сразу и промежуточные результаты, как только они станут доступны. В данном случае вы даете пользователям полезную информацию в необходимом виде. В YouTube, например, пользователи инициируют поиск по видео путем нажатия на кнопку Search. Если соединение с сетью низкое, то, прежде всего YouTube отображает сообщение “Loading...” с вращающимся индикатором активности, чтобы пользователи понимали, что поиск ведется в процессе. Затем YouTube отображает список результатов, в котором каждая строка наполнена текстом, таким как заголовок видео и рейтинг просмотров, а так же индивидуальное изображение бокса с пунктирной обводкой. Пока пользователь просматривают список названий видео, видео-превью заменяют пунктирные боксы, по мере загрузки.

Обдумайте предоставление панели возможностей, если данные сортируются непосредственно в различных категориях. Панель возможностей позволяет пользователям определять места расположения или правила поиска или фильтровать объекты по определенным критериям. Она содержит до четырех кнопок возможностей, каждая из которых представляет какую-либо категорию. Например, приложение Mail предоставляет панель возможностей, которая позволяет пользователям сфокусировать их поиск по категориям “От Кого”, “Кому” или по теме сообщения, или производить поиск, включая все поля. Предоставление этой панели помогает пользователям сфокусировать их поиск и может значительно уменьшить количество результатов. Чтобы узнать больше об использовании панели, смотрите “Score Bar” (стр. 143)

Entice and Inform with a Well-Written Description (Привлекайте и Информировать Хорошо Написанными Описаниями)

Ваше описание в App Store – это отличная возможность общения с потенциальными пользователями. В дополнение к описанию аккуратной работы вашего приложения, подчеркивайте те качества приложения, которые людям должны понравиться больше всего, придерживайтесь следующих правил:

Убедитесь, что откорректированы все возможные неправильные написания слов, грамматические и пунктуационные ошибки. И, хотя такие ошибки никого не беспокоят, у некоторых пользователей может сложиться негативное впечатление о качестве вашего приложения.

Сведите количество заглавных букв к минимуму. Отображаемое время от времени заглавные буквы могут привлечь внимание пользователей, но если каждая буква будет заглавной в каждом слове в описании – этот текст можно будет прочитать с трудом.

Обдумайте описание исправленных ошибок. Если новая версия вашего приложения содержит исправления ошибок, ожидаемых пользователями, было бы неплохо упомянуть об этом в вашем описании.

Be Succinct (Будьте Лаконичны)

Думайте как редактор газеты, старайтесь передать информацию в сжатом стиле заголовка. Когда ваш UI текст является коротким и прямым, пользователи могут осознать информацию быстро и просто. Определяйте наиболее важную информацию, выражайте ее кратко и отображайте ее в заметном виде – и пользователям не понадобится читать слишком много слов, чтобы отыскать то, что они ищут или определить, что делать дальше.

Давайте контроллерам короткие названия или используйте хорошо понятные символы, чтобы пользователи могли понять, что они делают с самого первого взгляда. При необходимости, используйте встроенные кнопки и иконки, описанные в “System-Provided Buttons and Icons” (стр 149); в качестве руководства по созданию своих собственных иконок, смотрите “Icons for Navigation Bars, Toolbars, and Tab Bars” (стр 170).

Use UI Elements Consistently (Используйте UI Элементы Должным Образом)

Пользователи ожидают, что стандартные окна и контроллеры выглядят и ведут себя должным образом во всем приложении.

Следуйте рекомендуемым способам использования стандартных элементов пользовательского интерфейса. В этом случае пользователи могут зависеть от их основного восприятия, что поможет им в обучении использованию вашего приложения. Кроме того, сделайте так, чтобы ваше приложение выглядело современно и правильным образом работало, даже если iOS меняет внешний вид и поведение этих стандартных окон или контроллеров.

Для приложений, в которых необходимы сильно вовлекающие задачи, такие как игры, логично создать полностью индивидуализированные контроллеры. Это потому что вы создаете уникальное окружение и открытие того, как взаимодействовать с этим окружением – это именно то, чего ожидают пользователи.

Избегайте радикального изменения внешнего вида контроллера, который предоставляет стандартное действие.

Если вы используете непривычные контроллеры для предоставления стандартных действий, пользователи потратят свое время на изучение того, как они работают и будут очень удивлены, если обнаружат, что ваши контроллеры делают то же самое, что и стандартные.

iOS предоставляет вам множество стандартных кнопок и иконок, использующихся во встроенных приложениях. Например, вы можете использовать те же самые иконки Refresh, Organize, Trash, Reply, и Compose (Обновить, Организовать, Мусор, Ответить и Создать), которые использует приложение Mail на iPhone и iPad.



Чтобы не сбивать с толку пользователей, никогда не используйте стандартные кнопки и иконки в каком-то ином значении. Убедитесь, что вы понимаете задокументированное значение стандартных кнопок или иконок; не оставляйте это на ваше восприятие их внешнего вида. Чтобы узнать больше об использовании элементов, предоставляемых системой, смотрите “System-Provided Buttons and Icons” (стр. 149).

В дополнение к преимуществу усиления пользовательского впечатления при использовании кнопок и иконок, предоставляемых системой, есть еще два других сильных преимущества:

- Уменьшение времени разработки приложения, потому что у вас нет необходимости создавать индивидуальную работу для передачи стандартных функций.
- Увеличение стабильности вашего пользовательского интерфейса, даже если в будущем iOS изменит внешний вид стандартных иконок. Другими словами, вы можете положиться на семантическое значение стандартной иконки, которое останется прежним, даже если внешний вид изменится.

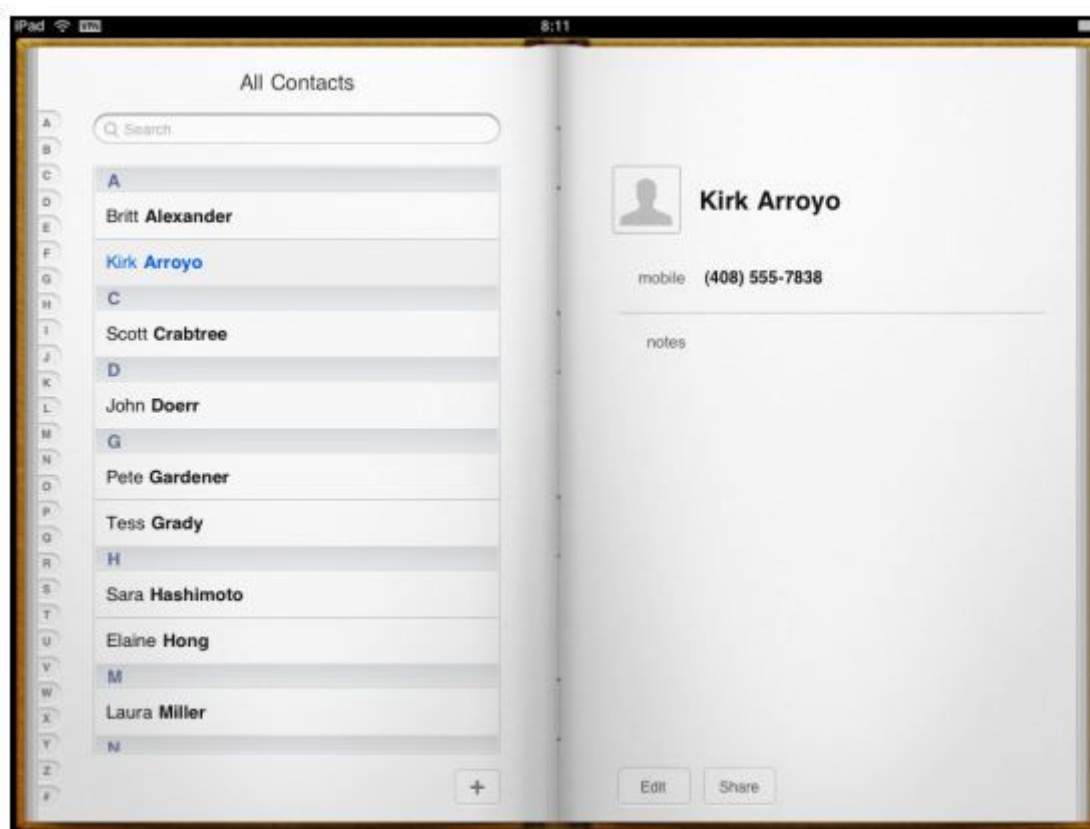
Interface Builder - делает простым использование кнопок, предоставленных системой, и применяет иконки, предоставленные системой, к вашим контроллерам. В качестве руководства смотрите информацию касательно внешнего вида иконок и кнопок в “iPhone OS Interface Objects” в Interface Builder User Guide (Пользовательском руководстве Interface Builder).

Если вы не можете найти системную кнопку или иконку, у которой есть необходимое вам значение, для определенной функции в вашем приложении, вы должны создать индивидуализированную кнопку или иконку. Руководство, которое поможет вам это сделать, вы можете увидеть в разделе “Icons for Navigation Bars, Toolbars, and Tab Bars” (стр. 170)

Consider Adding Physicality and Realism (Обдумайте Добавление Физичности и Реализма)

При необходимости добавьте реалистичность и физический объем в ваше приложение. Очень часто, чем приближеннее к жизни выглядит и ведет себя ваше приложение, тем проще для пользователей понять, как оно работает и тем больше они наслаждаются его использованием.

Например, пользователи действительно знают, как использовать настоящую адресную книгу, которая соотносится с приложением Contacts на iPad.



На iPhone пользователи знают, как работает приложение Voice Memos, и как его использовать, потому что оно предоставляет прекрасно отрендеренную картинку (микрофона) и реалистичные контроллеры.



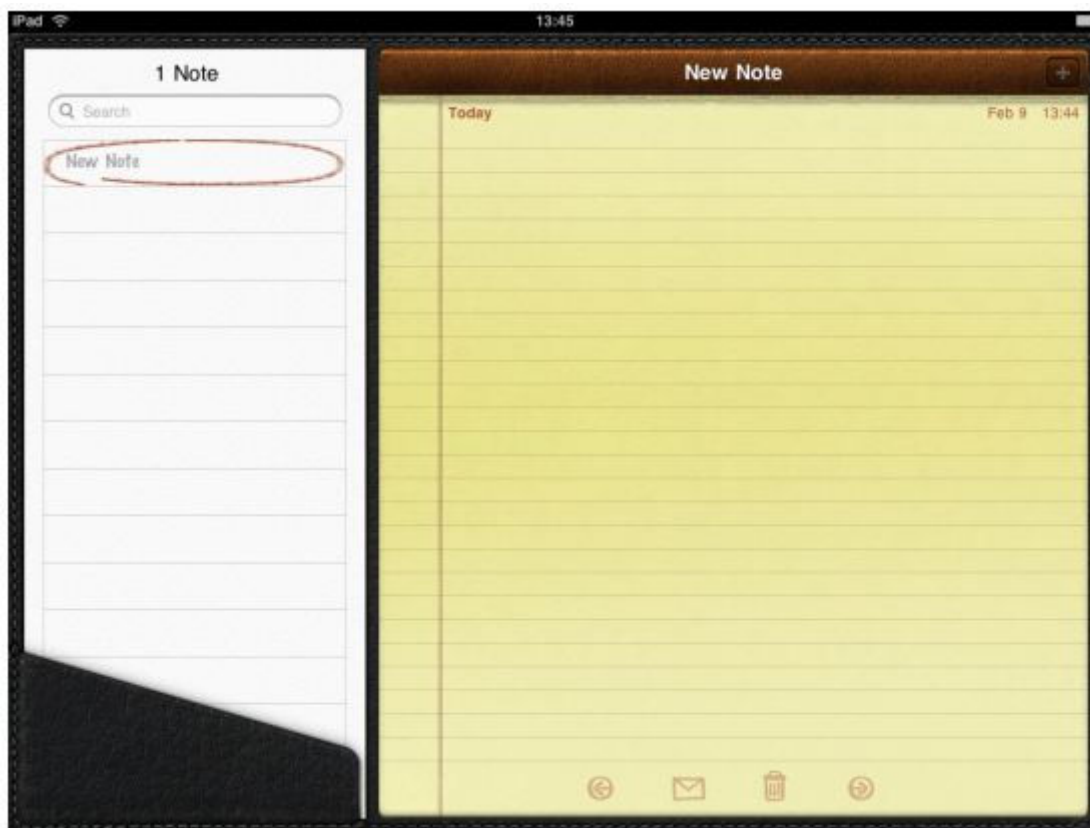
Думайте о том, что объекты и экраны, которые вы создаете - это возможность для коммуникации с пользователями и выражения сути вашего приложения. Не чувствуйте себя обязанным добиться доскональной аккуратности. Часто утрированное или улучшенное изображение чего-либо может выглядеть более реалистично и передавать большее значение, чем четкая схожесть с объектом.

Используйте необходимую анимацию для дальнейшего усиления реалистичности вашего приложения. В общем, более важно стремиться к соответствию в движении, чем к внешнему виду. Пользователи принимают художественную свободу во внешнем виде, но они могут почувствовать дезориентацию, когда видят движение, идущее наперекор физическим законам. Как можно точнее убедитесь, что ваши виртуальные окна и контроллеры копируют поведение физических объектов и контроллеров, которым они подражают. Убедительная анимация резко увеличивает пользовательское впечатление от вашего приложения, подобно физическому, существенный мир, в котором они хотят провести время.

Delight People with Stunning Graphics (Восхищайте Пользователей Умопомрачительной Графикой)

Богатая, прекрасная и привлекательная графика привлекает пользователей и награда не заставит себя ждать. Прекрасная художественная работа также помогает выстроить бренд вашего приложения в глазах пользователей. Устройства, основанные на iOS, являются витриной художественной работы вашего приложения, следовательно, вы должны обдумать привлечение к работе профессионального художника для создания первоклассной графики, которой будут восторгаться пользователи.

Обдумайте применение высококачественных или дорогих материалов. Если эффект дерева, кожи или металла должным образом подходит к вашему приложению, потратьте время, чтобы убедиться, что эти материалы выглядят реалистично и дорого. Например, приложение Notes предоставляет внешний вид отлично выделанной кожи и тщательной отделки швов.



При необходимости, создайте работу в высоком разрешении. В большинстве случаев, растягивание работы не рекомендуется в качестве долгосрочного решения. Вместо этого попытайтесь создать вашу работу в разрешении, большем, чем вам необходимо, и таким

образом вы сможете добавить глубину и детали, прежде чем уменьшать работу до нужного размера. Это работает особенно хорошо, когда разрешение оригинальной работы в несколько раз больше, чем то разрешение, которое вам необходимо. Затем, если вы также используете необходимый размер в вашем приложении по редактированию изображения, то у вас будет возможность сохранить файл уменьшенной работы в четком виде и уменьшить количество ретуши и постобработки, которую вам необходимо сделать.

Убедитесь, что изображение подключения и иконка приложения выполнены в высоком качестве.

В качестве руководства, смотрите “Custom Icon and Image Creation Guidelines” (стр. 157).

Уберите трудно кодируемые значения, которые идентифицируют размеры экрана. Это особенно важно, если вы хотите, чтобы ваше приложение работало на различных устройствах, основанных на iOS.

Handle Orientation Changes (Поддержка Смены Ориентации)

Пользователи достаточно часто планируют использовать свои iOS-устройства, в любой ориентации. Вам необходимо определить, как именно вы ответите на данное ожидание внутри функционала вашего приложения и задач, которые открывает ваше приложение.

Во всех ориентациях поддерживайте фокус на основном контенте. Это ваш наивысший приоритет.

Пользователи используют ваше приложение, чтобы просматривать и взаимодействовать с контентом, который им интересен. Изменение фокусировки на контенте в различных ориентациях может вызвать у пользователей ощущение, что они теряют контроль над приложением.

Подумайте дважды, прежде чем отказаться от работы вашего приложения во всех ориентациях. Пользователи ожидают использовать ваше приложение в различных ориентациях, и самый наилучший вариант - удовлетворить это ожидание. Пользователи iPad, в частности, ожидают использовать ваше приложение в любой из ориентаций, в зависимости от того, как они держат устройство. Но в некоторых случаях приложению необходимо работать только в горизонтальной или только в вертикальной ориентации.

Если очень важно, чтобы ваше приложение работало только в одной ориентации, вы должны:

- **Запускать ваше приложение в той ориентации, которую вы поддерживаете, вне зависимости от текущей ориентации устройства.** Например, если ваше приложение (игра или приложение для просмотра медиа) работает только в горизонтальной ориентации, необходимо, чтобы приложение запустилось в горизонтальной ориентации, даже если устройство сейчас находится в вертикальной ориентации. То есть, если пользователи запустят ваше приложение в вертикальной ориентации, они будут знать, что необходимо перевернуть устройство в горизонтальную ориентацию, чтобы просмотреть контент.

- **Избегайте отображения UI-элемента, который сообщает пользователям о необходимости повернуть устройство.** Запуск приложения в поддерживаемой вами ориентации очень четко сообщает пользователям, что необходимо перевернуть устройство, без добавления бесполезной неразберихи в ваше UI.

- **Поддерживаете оба варианта ориентации.** Например, если ваше приложение работает только в горизонтальной ориентации, у пользователей должна быть возможность использовать его, когда кнопка Home находится как справа, так и слева. И, если пользователи переворачивают устройство на 180 градусов, в то время как используют ваше приложение, наилучшим ответом будет переворот вашего контента на 180 градусов.

Если ваше приложение интерпретирует изменения в ориентации устройства, как пользовательский ввод данных, то вы можете управлять вращением, которое необходимо для данного приложения. Например, если ваше приложение - это игра, которая позволяет пользователям передвигать элементы игры, вращая устройство, то вы не можете отвечать на вращение устройства вращением экрана. В этом случае, вы должны запустить приложение в том варианте ориентации, которая необходима, и позволить пользователям переключать между вариантами до тех пор, пока они не начнут основную задачу приложения. Затем, как только пользователи начнут выполнять основную задачу приложения, вы можете начать отвечать на движения устройства особым для данного приложения способом.

Используйте преимущество одношагового изменения ориентации, чтобы предоставить более мягкий переход. Однако, если шаблон вашего экрана слишком сложный, то, возможно, вам лучше выбрать вместо этого кросс-фэйд переход, когда пользователи изменяют ориентацию устройства. "Handling View Rotations" в UIViewController Class Reference объясняет как использовать одношаговый процесс в вашем коде.

Обращайте внимание на значения акселерометра. Чтобы узнать больше об этих данных и их получение, смотрите Core Motion Framework Reference. Если это необходимо, ваше приложение должно отвечать на все изменения в ориентации устройства.

На iPhone предчувствуйте то, что пользователям необходимо, чтобы вы отвечали на изменения в ориентации устройства. Пользователи часто разворачивают свои устройства в горизонтальную ориентацию, потому что они хотят "увидеть больше". Если вы отвечаете только увеличением вашего контента, то вы не оправдываете ожидания пользователей. Вместо этого, вы должны отвечать переделкой линий и текста, и, если необходимо, переделать шаблон пользовательского интерфейса, чтобы на экране помещалось больше контента.

На iPad старайтесь удовлетворить пользовательские ожидания так, чтобы приложения могли работать в обеих ориентациях. Большой экран iPad подталкивает пользовательское желание перевернуть устройство в горизонтальную ориентацию, чтобы "увидеть больше". И, так как пользователи не обращают большого внимания на меньшую сторону устройства или место расположения кнопки Home, они не рассматривают устройство как что-то имеющее ориентацию по умолчанию. Недостаточная обеспокоенность ориентацией устройства по умолчанию приводит к тому, что пользователи ожидают от приложения хорошей работы в той ориентации устройства, в которой они находятся в данный момент времени. Как только это реализовано, ваше приложение должно усиливать взаимодействие пользователей с iPad с любой стороны, предоставляя отличный опыт во всех ориентациях.

Следуйте данному руководству в дизайне вращения iPad:

- Обдумайте изменения в отображении вспомогательной информации или функционала. Хотя вы и должны убедиться, что наиболее важный контент всегда остается в фокусе, вы можете ответить на разворот изменением вторичного контента.

В приложении Mail на iPad, например, списки аккаунтов и писем являются вторичным контентом (основной контент- это выбранное сообщение). В горизонтальной ориентации вторичный контент отображается на левой панели split view (разделительное окно); в вертикальной ориентации он отображается на popover.

Или обдумайте iPad игру, отображающую прямоугольное поле игры в горизонтальной ориентации. В вертикальной ориентации игра должна быть перерисована, чтобы хорошо подходить к размеру экрана. Вместо вертикального растягивания игрового поля, закрывая пространство или оставляя его пустым, игра должна отображать дополнительную информацию или объекты в дополнительном пространстве.

- Избегайте неуместных изменений в структуре. Насколько возможно, предоставляйте впечатление устойчивости приложения во всех ориентациях. Соизмеримое впечатление во всех ориентациях позволяет пользователям поддерживать использование структуры при вращении устройства. Например, если ваше iPad приложение отображает картинки,

расставленные по сетке в горизонтальной ориентации, то нет необходимости отображать ту же самую информацию в списке, когда приложение находится в вертикальной ориентации (хотя вы можете отрегулировать размер сетки).

- **По возможности, избегайте деформирования информации и перекрытия текста при развороте.** Стремитесь поддерживать одинаковый формат во всех ориентациях. Особенно, если пользователи читают текст, очень важно избегать того, чтобы текст изменял свое местоположение при повороте устройства.

Если же реформирования информации нельзя избежать, используйте анимацию, чтобы помочь пользователям отследить изменения. Например, если вы должны добавить или удалить столбец текста в другой ориентации, вы можете выбрать скрытие передвижения столбцов и просто сделать анимацию скрытия в новом расположении. Чтобы помочь себе в дизайне необходимого поведения при развороте, подумайте о том, чего бы вы ожидали в поведении вашего контента, если бы вы физически взаимодействовали с ним в реальном мире.

- **Предоставьте уникальную картинку соединения для каждой ориентации.** Когда каждая ориентация имеет уникальную картинку соединения, пользователи легко включаются в работу с приложением, вне зависимости от текущей ориентации устройства. В сравнении с экраном на iPhone, на iPad экран поддерживает все ориентации, и таким образом, пользователи могут запустить ваше приложение в той же ориентации, в которой они вышли из предыдущего приложения.

Make Targets Fingertip-Size (Создавайте Цели Размером с Кончики Пальцев)

Размер экрана на устройстве, основанном на iOS, может варьироваться, но средний размер кончика пальца - нет. Вне зависимости от того, на каком устройстве работает ваше приложение, следуя данному руководству, убедитесь, что пользователи могут комфортно использовать ваше приложение.

Задайте всем нажимаемым элементам в вашем приложении размер около 44x44 пикселя. Встроенное приложение iPhone-Калькулятор - это хороший пример контроллеров, которые подходят под зону нажатия пальцем.



Если вы создадите контроллеры меньшего размера или разместите их слишком близко друг к другу, пользователи должны будут внимательно прицеливаться, чтобы нажать на них, и велика вероятность, что они нажмут не на тот элемент. В результате приложение становится менее привлекательным или даже совершенно невозможным в использовании. Например, игра, в которой контроллеры маленького размера и расположены слишком близко друг к другу, заставляет пользователей концентрироваться на интерфейсе, вместо игры.

Use Subtle Animation to Communicate (Используйте Едва Различимую Анимацию для Коммуникации)

Анимация - является отличным способом эффективного взаимодействия, так как она не прерывает выполнение пользователями своих задач и не замедляет их работу. Едва различимая и созданная должным образом анимация может:

- Взаимодействовать со статусом
- Предоставлять полезную обратную связь
- Усиливать смысл прямой манипуляции
- Помочь пользователям визуализировать результаты своих действий

Добавляйте анимацию очень осторожно, особенно в приложениях, которые не предоставляют увлекательного впечатления. В приложении, которое сфокусировано на серьезных или продуктивных задачах, анимация, которая кажется экспрессивной или неуместной, может затруднить ход работы приложения, уменьшить производительность и отвлечь пользователей от задачи.

Сделайте анимацию согласующейся со встроенными приложениями, когда это необходимо. Пользователи привыкли к использованию чуть заметных анимаций во встроенных iOS приложениях. По факту, большинству пользователей нравятся мягкие переходы между окнами, быстрый отклик на изменение ориентации устройства и реалистичные переворот и скроллинг - это ожидаемая часть впечатления от работы iOS. Если

вы не создаете приложение, которое предоставляет глубокое впечатление, такое как игра, индивидуализированная анимация должна быть совместима со встроенными анимациями.

Используйте анимацию в вашем приложении согласованно. Так же как и с другими типами индивидуализирования, важно использовать индивидуализированную анимацию согласованно со всем приложением, чтобы пользователи могли положиться на опыт, который они получают с вашим приложением.

Support Gestures Appropriately (Поддержка Необходимых Жестов)

Избегайте ассоциации стандартных жестов с действиями, отличными от тех, которые пользователи уже знают. Например, позаботьтесь, чтобы избежать переопределения значения “пролистать сверху вниз” (“swipe down from the top”) жеста, который вытаскивает Центр Уведомлений (Notification Center).

Очень важно избегать создания индивидуализированных жестов, чтобы использовать их для действий, которые пользователи уже ассоциируют со стандартными жестами.

Используйте сложные жесты в качестве варианта, для ускорения задачи, а не только как способ предоставить задачу. Хотя большинство пользователей знают более сложные стандартные жесты, такие как “swipe” или “pinch open”, эти сложные жесты не применяются повсеместно.

Например, при просмотре списка сообщений в приложении Mail, пользователи удаляют сообщение жестом “swipe”, а затем нажатием на кнопку Delete в строке предпросмотра сообщения. Пользователи могут открыть кнопку Delete двумя разными способами:

- Нажать на кнопку Edit на панели навигации, которая открывает контроллер удаления в каждой строке предпросмотра. Затем нажать на контроллер удаления в определенном ряду предпросмотра, чтобы отобразить кнопку Delete для данного сообщения. Этот метод включает в себя дополнительный шаг, но очевиден, так как для него необходимо только нажать и начать с четко определенной кнопки Edit.
- Сделав жест swipe через строку предпросмотра, чтобы открыть кнопку Delete для данного сообщения. Этот метод быстрее, но для него необходимо, чтобы пользователь выучил и запомнил более специфичный жест swipe.

Попробуйте убедиться, что всегда есть простой, четкий путь предоставления действия, даже если это означает одно или два дополнительных нажатия. Простые жесты позволяют пользователям сфокусироваться на контенте и впечатлениях, а не на взаимодействии.

В общем, избегайте определения новых жестов. Когда вы предоставляете новые жесты, пользователи должны предпринять попытку, чтобы обнаружить и запомнить их. Основное исключение из данной рекомендации - это приложение, которое предоставляет увлекательный опыт, в котором индивидуализированные жесты могут быть необходимы. Например, в приложении по созданию документа, в котором необходимо сделать круговой жест, чтобы открыть кнопку Delete в строке таблицы - это может сбивать с толку и сложно для использования. Но в игре, скорее всего, понадобится круговое движение, чтобы “кружить” какую-либо часть игры.

Убедитесь, что движения, которые вы используете, имеют смысл в контексте функционала вашего приложения и ожиданий пользователей. Если, например, в вашем приложении

имеется важная задача, которая встречается достаточно часто и пользователям необходимо быстро ее выполнять, то вам, скорее всего, необходимо использовать только стандартные жесты. Но если ваше приложение содержит в себе реалистические контроллеры, которые определяют специфическое использование, или предоставляет среду, которую пользователи ожидают изучить, то индивидуализированные жесты могут быть наилучшим решением. (Для получения более подробной информации о стандартных жестах, смотрите “Apps Respond to Gestures, Not Clicks” (стр 10).)

Для iPad обдумайте использование жестов увеличения картинки. Большой экран iPad предоставляет отличную возможность для использования жестов увеличения, включая жесты, которые могут произвести несколько человек. Хотя сложные жесты и не являются необходимыми для каждого приложения, они могут увеличить впечатление от приложения, с которым пользователи проводят много времени, такого как игра или среда по созданию контента. Всегда помните, что нестандартные жесты не так легко обнаружить, и они должны использоваться только для предоставления специфического действия.

Ask People to Save Only When Necessary (Запрашивайте у Пользователей Сохранение Только в Случае Необходимости)

Пользователи должны быть четко уверены, что их работа всегда находится в сохранности, кроме тех случаев, когда они сами явно отменяют ее или удаляют. Если ваше приложение помогает пользователям создавать или редактировать документы, убедитесь, что они не должны четко применять функцию сохранения. iOS-приложения должны брать ответственность на себя за сохранение введенных пользовательских данных время от времени и при открытии других документов или выходе из приложения.

Если основной функцией вашего приложения не является создание контента, но вы позволяете пользователям переключаться между просмотром информации и редактированием ее, имеет смысл давать запрос на сохранение изменений. В этом случае применяется предоставление кнопки Edit в окне, которое отображает информацию. Когда пользователи нажимают на кнопку Edit, вы можете заменить ее на кнопку Save, и добавить кнопку Cancel. Трансформация кнопки Edit помогает напомнить пользователям, что они находятся в режиме редактирования и, возможно, необходимо сохранить изменения, а кнопка Cancel дает им возможность выйти без сохранения изменений.

Для iPad сохраняйте информацию, которую пользователи вводят в popover (если только они не отменили свою работу), потому что они могут покинуть popover неосознанно. Для более подробного руководства, касательно использования popover, смотрите “Popover (iPad Only)” (стр 108).

Make Modal Tasks Occasional and Simple (Очень Редко Создавайте Модальные Задачи и Делайте Их Простыми)

При возможности, минимизируйте ситуации, когда пользователи должны находиться в модальном режиме для предоставления задачи или предоставить ответ. iOS-приложения должны позволять пользователям взаимодействовать нелинейными способами (in nonlinear ways). Модальность предотвращает данную свободу, прерывая рабочий процесс пользователей и заставляя их выбирать определенный путь.

Модальность наилучшим образом подходит к использованию, когда:

- Критично привлечь внимание пользователя.
- Задача должна быть завершена (или ясно отвергнута), чтобы предотвратить сохранение пользовательских данных в недвусмысленном состоянии.

Пользователи предпочитают выполнять замкнутую подзадачу в модальном окне, потому что контекст сдвигается четко и только на определенное время. Но если подзадача является слишком сложной, пользователи могут потерять основную задачу, которую они отложили, из вида в тот момент, когда они входят в модальное окно.

Этот риск увеличивается, когда модальное окно увеличено на весь экран и когда оно содержит множество зависимых окон или состояний.

Поддерживайте внешний вид модальных задач короткими и узкофокусированными.

Не надо предоставлять пользователям модальное окно как мини-приложение внутри вашего приложения. Особенно остерегайтесь создания модальной задачи, которая затрагивает иерархию окон, потому что пользователи могут попросту потеряться и забыть свои собственные шаги. Если модальная задача содержит подзадачи в отдельных окнах, убедитесь, что вы предоставляете пользователям один четкий путь через иерархию и избегаете зацикливания.

Всегда предоставляйте четкий и безопасный способ выхода из модальной задачи.

Пользователи должны всегда иметь возможность предугадать судьбу своей работы при отключении модального окна.

Если задаче необходима иерархия модальных окон, убедитесь, что пользователи понимают что происходит, если они нажимают на кнопку Done в окне, находящемся под самым верхним уровнем. Проверяйте задачу, чтобы решить, должна ли кнопка Done в окне нижнего уровня завершать только ту часть задачи, которая соответствует данному окну, или завершать всю задачу целиком. По мере возможности, избегайте добавления кнопки Done на зависимые окна, потому что обычно это сбивает с толку.

Start Instantly (Начинайте Сразу)

Часто говорят, что пользователи тратят 1-2 минуты на оценку нового приложения. Если вы займете это время наиболее важным и интересным контентом, вы будете на пике внимания новых пользователей, а также все пользователи получат превосходное впечатление.

Отображайте изображение запуска (launch image), которое имеет четкое сходство с первым экраном приложения. Эта практика уменьшает восприятие времени соединения с вашим приложением.

Избегайте изображения окна About или рекламного экрана. В общем, попытайтесь избежать предоставления какого-либо типа начального впечатления, прежде чем пользователи воспользуются вашим приложением.

На iPhone определите необходимый стиль статус-панели. В общем, вы хотите, чтобы статус-панель соотносилась с остальной частью UI вашего приложения.

Включайте приложение в необходимой ориентации по умолчанию. На iPhone ориентация по умолчанию - вертикальная; на iPad ориентация по умолчанию - это текущая ориентация устройства. Если вы планируете, что ваше приложение должно использоваться только в горизонтальной ориентации, пусть приложение включается в горизонтальной ориентации, вне зависимости от текущей ориентации устройства и позвольте пользователям перевернуть устройство в горизонтальную ориентацию, если это необходимо.

Примите во внимание, что приложение, работающее только в горизонтальной ориентации, должно поддерживать оба варианта горизонтальной ориентации - либо кнопка Home располагается справа, либо слева. Если физически устройство уже находится в горизонтальной ориентации, приложение, которое работает только в горизонтальной

ориентации, должно запуститься в этой ориентации, кроме тех случаев, когда есть существенная причина не делать этого. С другой стороны, приложение, которое работает только в горизонтальной ориентации, по умолчанию должно запускаться в горизонтальной ориентации, где кнопка Home находится справа.

Избегайте запроса у пользователей на предоставление информации на установку.

Вместо этого следуйте данным инструкциям:

- **Сфокусируйте свое решение на том, что необходимо 80% ваших пользователей.** Когда вы это делаете, большинству пользователей нет необходимости предоставлять настройки, потому что ваше приложение уже установлено на то поведение, которое они ожидают. Если есть какой-либо функционал, который может быть необходим только некоторому количеству пользователей, или который большинству пользователей может быть необходим только однажды, не обращайтесь на это внимание.
- **Получайте как можно больше информации из других источников.** Если вы можете использовать какую-либо информацию, которую пользователи предоставляют во встроенных приложениях или настройках устройства, то запросите систему на использование этих данных; не просите пользователей вводить их снова.
- **Если вам необходимо запросить информацию о настройке, подскажите пользователям ввести их внутри вашего приложения.** Затем, как можно быстрее, сохраните эту информацию (возможно в настройках вашего приложения). В этом случае пользователей никто не заставляет выходить из вашего приложения и открывать Настройки, прежде чем они смогут получить шанс насладиться вашим приложением. Если пользователям необходимо позже внести изменения в эту информацию, они могут в любое время попасть в настройки вашего приложения.

Задерживайте вход в приложение как можно дольше. В идеале, пользователи должны иметь возможность перемещаться по большей части приложения, чтобы понять, что они могут сделать с ним, до входа в свой аккаунт. Если вы будете просить пользователя войти в свой аккаунт до непосредственного использования приложением, им может показаться, что запуск приложения слишком долгий.

Восстанавливайте приложение в то состояние, в котором оно было, когда пользователь последний раз остановил использование приложения. Пользователи не должны запоминать путь, по которому они достигли предыдущего места в вашем приложении.

Важно: Не говорите пользователям перезагрузить или снова включить устройство после установки вашего приложения, потому что перезагрузка забирает дополнительное время и может показаться, что приложение очень трудно использовать. Если ваше приложение использует большое количество памяти или у вас есть какие-то другие проблемы, которые не дают нормально работать устройству без перезагрузки, вам необходимо указать эти проблемы. Для изучения руководства по разработке хорошо работающего приложения, смотрите “Using Memory Efficiently” в iOS Application Programming Guide.

Always Be Prepared to Stop (Всегда Будьте Готовы Остановиться)

iOS-приложение завершает свою работу, как только пользователи нажимают кнопку Home, чтобы открыть другое приложение или использовать функции устройства, такие как звонок по телефону. В частности, пользователи не нажимают на кнопку закрытия в приложении и не

выбирают Выход из меню. Чтобы составить хорошее пользовательское впечатление во время прекращения работы приложения, iOS-приложение должно:

- **Сохранять пользовательские данные, как только это возможно и так часто**, как это возможно, потому что выход или временное уведомление может случиться в любое время.
- **Сохранять текущее состояние при выходе из приложения** с максимально возможным уровнем детализации, чтобы пользователи не потеряли контент, когда они снова включают приложение. Например, если ваше приложение отображает данные скроллинга, сохраните текущую позицию прокрутки.

Don't Quit Programmatically (Не Выходите Программным Образом)

Никогда не делайте выход из приложения программным способом, потому что пользователи обычно интерпретируют это как обвал приложения. Однако, если внешние причины заставляют работать ваше приложение не так, как планировалось, вы должны сказать об этом вашим пользователям и объяснить, что они должны с этим сделать. В зависимости от того, как работает ваше приложение, у вас есть два пути.

Отобразите привлекающий внимание экран, который описывает проблему и предлагает ее решение. Этот экран предоставляет обратную связь с пользователем и объясняет ему, что ничего страшного не случилось с вашим приложением. Это дает пользователю способность управлять ситуацией – захочет ли он решить проблему и продолжить использование вашего приложения или нажать кнопку Home и открыть другое приложение.

Если какие-либо функции вашего приложения не работают, отобразите либо экран, либо предупреждение, когда пользователи активируют эти функции. Отображайте предупреждение только тогда, когда пользователи пытаются получить доступ к функции, которая не доступна.

If Necessary, Display a License Agreement or Disclaimer (При Необходимости Отобразите Пользовательское Соглашение или Отказ)

Если вы предоставляете конечному пользователю пользовательское соглашение (или EULA) вместе с вашим iOS-приложением, App Store отображает его, чтобы пользователи смогли его прочитать перед использованием.

Если это возможно, избегайте запроса к пользователю о вашем пользовательском соглашении, когда он впервые запускает ваше приложение. Без отображения соглашения пользователи могут наслаждаться вашим приложением без задержки. Однако, хотя это и предпочтительно для пользовательского впечатления, это не является универсальным во всех случаях. Если вам необходимо отобразить пользовательское соглашение внутри вашего приложения, делайте это через гармонизацию с вашим пользовательским интерфейсом и сведите к минимуму беспокойство пользователя по этому поводу.

По возможности предоставьте форму отказа внутри описания вашего приложения или Пользовательского Соглашения. Пользователи могут увидеть форму отказа в App Store, и вы сможете создать баланс между тем, что необходимо вашему бизнесу и тем, что необходимо пользователю.

For iPad: Enhance Interactivity (Don't Just Add Features) (Для iPad: Усиление Интерактивности (Не Добавляйте Функции))

Самые лучшие iPad-приложения передают пользователям инновационные пути взаимодействия с контентом путем предоставления четких задач.

Уменьшите попытки добавления функций, которые напрямую не относятся к основной задаче. Вместо этого изучайте способы, которые позволяют пользователям увидеть и взаимодействовать больше. В частности, вы не должны рассматривать большой iPad-экран как приглашение к возвращению всего того функционала, который вы обрезали в iPhone версии вашего приложения.

Чтобы заставить ваше приложение твердо держаться на ногах, сконцентрируйтесь на путях расширения пользовательского впечатления без разбавления основной задачи дополнительными функциями. Например:

- Приложение-читалка для книг, которое позволяет пользователям читать книги и отслеживать список прочитанного, может произвести достаточно сильное впечатление от прочтения на большом экране. Вместо того, чтобы заставлять пользователей переходить на другой экран для управления списком книг, приложение может положить этот список на ротор и позволить пользователям копировать туда любимые тексты. Приложение также может позволить пользователям добавлять закладки и аннотации к тексту, и помочь им распространять списки среди других пользователей или сравнивать их прогресс, вместо того, чтобы просто хранить списки.
- Игра – летный симулятор может дать возможность использования полупрозрачной бортовой системы самолета поверх основного меню. Пользователь может нажимать на реалистичные контроллеры выпуска снарядов, чтобы стрелять во врага или видеть, где он находится на карте.
- Игра – симулятор футбола может отображать более реалистичное игровое поле или более детальных персонажей, а также позволить пользователям управлять своей командой и индивидуализировать персонажей. Она также может позволить пользователям увидеть информацию о персонажах, не покидая окна игрового поля. В конце концов, приложение может позволить игру в режиме мультиплеера, когда двое пользователей могут выставить друг против друга свои команды.
- Приложение, которое помогает в написании сценария, может предоставить способы переключения между окном сюжета и окном персонажа, не покидая основного содержания. Писатели могут переключаться между этими окнами, чтобы проверить детали, описанные в основном окне.

For iPad: Reduce Full-Screen Transitions (Для iPad: Уменьшение Переходов с Полного Экрана)

Четко ассоциируйте визуальные переходы с изменяемым контентом. Вместо перехода на полностью новый экран с какими-то встроенными изменениями в информации, попытайтесь обновить только те области UI, которые необходимы. Основное правило – используйте переход на отдельные окна и объекты, а не на экран. В большинстве случаев, «переворот» целого экрана не рекомендуется.

Когда вы предоставляете меньшее количество полноэкранных переходов, ваше iPad-приложение выглядит визуально стабильным, что позволяет пользователям отслеживать свои

задачи. Вы можете использовать UI-элементы, такие как split view и popover для уменьшения необходимости в полноэкранных переходах.

For iPad: Restrain Your Information Hierarchy (Для iPad: Сдерживайте Иерархию Вашей Информации)

Используйте большой iPad-экран и специфические UI-элементы для iPad, чтобы предоставить пользователям больше информации в одном месте. Хотя вы и не должны вмещать слишком много информации на экран, вы также должны предотвратить такую ситуацию, при которой пользователи будут чувствовать, что им необходимо посетить множество различных экранов, чтобы найти то, что им необходимо.

В общем, сфокусируйте на основном экране самый важный контент и предоставьте дополнительную информацию или инструменты на вспомогательных окнах, таких как popover. Это позволяет пользователям простой доступ к функциональности, которая им необходима, без необходимости покидать контекст основной задачи.

С большим iPad-экраном и UI-элементами, такими как разделительное окно и поповер, вы имеете альтернативу структуре большинства iPhone-приложений – один-уровень-на-экран. (Для более четкого руководства по использованию этих элементов смотрите “Split View (iPad Only)” (стр 112) and “Popover (iPad Only)” (стр 108).)

Используйте панель навигации на правой панели в split view, чтобы позволить пользователям углубляться уровни категории, отображенной на левой панели. **Split view** выравнивает иерархию вашей информации как минимум на один уровень, потому что два уровня всегда находятся на экране в одно и тоже время. Например, приложение Настройки отображает настройки устройства и приложения, используя панель навигации в правой панели split view.



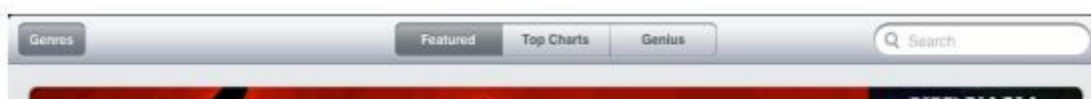
Используйте панель навигации на левой панели split view, чтобы позволить пользователям проходить через достаточно неглубокую иерархию. Затем отобразите более специфическую информацию (узел иерархии) на правой панели. Это также упрощает вашу иерархию, отображая два уровня на экране в одно и то же время. Например, приложение Mail в горизонтальной ориентации использует этот дизайн для отображения иерархии аккаунтов, почтовых ящиков и списка сообщений на левой панели, и отдельных сообщений в правой панели.



Используйте popover, чтобы предоставить действия или предоставить инструменты, которые воздействуют на объекты, располагающиеся на экране. Поповер может временно отображать данные действия и инструменты поверх текущего экрана. Например, Напоминание использует поповер, чтобы предоставить пользователю простой способ добавления деталей в пункт напоминания, который он создал.



Используйте сегментный контроллер в панели инструментов для отображения различных перспектив в контенте или различных категорий информации. В этом случае вы можете обеспечить доступ к перспективам и категориям из одной панели в верхней (или нижней) части экрана. В качестве руководства см. "Toolbar" (стр. 104) и "Segmented Control" (стр. 145). Например, iTunes использует сегментный контроль в верхней части панели инструментов, чтобы обеспечить различный подход к содержанию категории.



Используйте панель ярлыков для отображения различных категорий информации, или различных режимов приложения. В iPad-приложениях панель ярлыков используется больше как фильтр или переключатель категорий, чем переключатель между режимами. Например, iTunes использует панель ярлыков (tab bar) для предоставления доступа к различным категориям медиа.



По возможности избегайте использования панели ярлыков для переключения по полностью различным экранам лучше уменьшить полно-экранные переходы между экранами на iPad.

For iPad: Consider Using Popovers for Some Modal Tasks (Для iPad: Обдумайте Использование Popover для Некоторых Модальных Задач)

Поповер и модальные окна одинаковы в том смысле, что пользователи обычно не могут взаимодействовать с основным окном, в то время как поповер или модальное окно открыты. Но модальное окно всегда модально, тогда как поповер может использоваться двумя различными способами:

- **Модально** - в этом случае поповер затемняет область экрана вокруг него и для него необходимо явное его закрытие. Это поведение очень похоже на поведение модального окна, но внешний вид поповера имеет тенденцию к созданию более легкого впечатления.
- **Немодально** - в этом случае поповер не затемняет область экрана вокруг него, и пользователи могут нажать в любом месте экрана за пределами его границ (включая контроллер, который открывает поповер), чтобы закрыть его. Это поведение делает немодальный поповер похожим на любое другое окно приложения.

В дополнение у рорровер всегда есть стрелка, которая указывает на контроллер или область, где пользователь нажимает, чтобы его открыть. Эта визуальная привязка помогает пользователям помнить предыдущий контекст. Это также заставляет модальный поповер выглядеть более временным, чем модальное окно, которое отображается поверх экрана без какой-либо индикации, откуда оно было открыто.

Если вы используете модальные окна, чтобы отображать замкнутые задачи в вашем iPhone-приложении, то лучше вместо этого использовать рорровер. Чтобы помочь себе решить, когда его лучше всего использовать, определите для себя следующие вопросы:

- **Необходимо ли вашей задаче более одного типа ввода данных?** Если да, используйте рорровер.

Хотя клавиатура может быть поддержана как рорровер, так и модальным окном, поповер наиболее подходит для отображения пикера или списка опций.

- **Необходимо ли, чтобы для решения вашей задачи пользователи углублялись в иерархию окон?** Если да, используйте поповер.

Форма рорровер подходит лучше для отображения множества страниц, потому что таким образом создается меньшая вероятность спутать его с основным окном приложения.

- **Возможно, пользователям будет необходимо что-то сделать на основном окне приложения, прежде чем закончить задачу?** Если да, используйте немодальный рорровер.

Так как пользователи могут видеть основное окно под немодальным рорровер и могут выйти из него нажатием на основное окно, вы должны позволить им задержать выполнение задачи на рорровер и сразу вернуться на него вскоре.

● **Находится ли задача достаточно глубоко в иерархии и представляет ли она одну из основных функций приложения?** Если да, возможно вам лучше использовать модальное окно.

Большой сдвиг контекста модального окна помогает пользователям оставаться сфокусированным на задаче, пока они ее не закончат. Большой размер экрана большинства стилей модальных окон делает проще предоставление множества вводной информации. Если, с другой стороны, задача представляет собой важную часть функционала приложения, но она не находится в глубине иерархии, модальный поповер может быть куда лучшим выбором. Это потому что более легкий визуальный вес поповера может лучше подходить к часто предоставляемой задаче.

● **Задача предоставляется только один раз или редко, в качестве задачи настройки?** Если да, обдумайте использование модального окна.

Пользователи не беспокоятся о том, что они остаются в текущем контексте, если они выполняют задачу только один раз или очень редко.

Существует также множество других способов использования popover, таких как предоставление дополнительных инструментов (в качестве более полного руководства по использованию, смотрите “Popover (iPad Only)” (стр. 108)). Также iPad приложения отображают списки действий внутри поповера (в качестве более подробной информации смотрите “Action Sheet” (стр. 129)).

Если вы решили использовать модальное окно, убедитесь, что вы прочитали о различных презентационных стилях, которые вы можете использовать (они описаны в “Modal View” (стр. 132)). В своем iPad-приложении вы можете выбрать презентационный стиль, который наилучшим образом подходит к модальной задаче, которую вам необходимо предоставить.

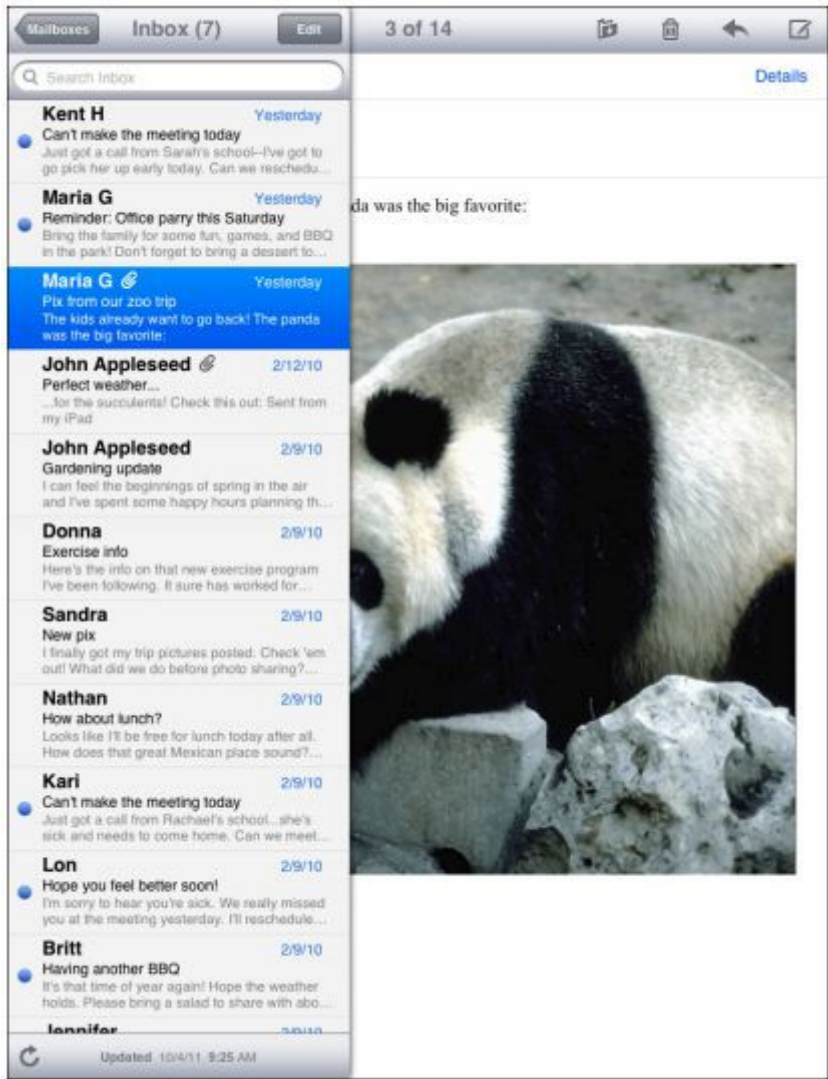
For iPad: Migrate Toolbar Content to the Top (Для iPad: Переместите Контент панели Инструментов Наверх)

Если в вашем iPad-приложении есть панель Инструментов, подумайте о переносе ее на верхнюю часть экрана, вместо того, чтобы оставлять ее внизу. Благодаря дополнительной ширине экрана iPad, вы сможете предоставить весь функционал своей панели Инструментов на одной панели сверху. Это дает вам больше вертикального пространства для вашего основного контента.

Например, приложение Mail на iPhone использует панель инструментов в нижней части экрана, чтобы предоставить пользователям доступ к обновлению, организации, удалению и другим действиям во время просмотра сообщения.



В приложении Mail на iPad переносится все, кроме одного из действий, на панель Инструментов, располагающуюся над сообщением. Контроллер Обновления располагается в нижней части списка почтового ящика.



iOS Technology Usage Guidelines

Руководство по Использованию iOS Технологий

iOS предоставляет множество великолепных технологий, которые ценятся пользователями, таких как: многозадачность, копирование, вставка, два вида локальных уведомлений (поставленные в задачу внутри вашего приложения и доставленные iOS на устройство), а также push-уведомления (пересылаемые в ваше приложение с удаленного сервера).

С точки зрения пользователя эти технологии являются частью iOS-оболочки, но разработчики знают, что требуется большое внимание, чтобы убедиться, что эти особенности хорошо сочетаются с пользовательским восприятием приложений.

iCloud Storage (Хранилище iCloud)

Хранилище iCloud помогает пользователям получить доступ к интересующему их содержимому независимо от того, какое устройство в настоящее время они используют. Если вы используете хранилище iCloud в своем приложении, пользователи могут использовать различные экземпляры приложения на различных устройствах для просмотра и редактирования их содержимого без выполнения явной синхронизации. Для того чтобы обеспечить это пользователям, вполне вероятно, вам придется пересмотреть способы хранения и получения доступа и предоставления информации (в частности, созданного пользователями контента) в вашем приложении. Чтобы изучить советы относительно способов структурирования вашего приложения таким образом, чтобы оно хорошо взаимодействовало с iCloud, а также программирования интерфейса iCloud, см. “iCloud Storage”.

Одним из основополагающих аспектов работы пользователей с хранилищем iCloud является прозрачность: в идеале, пользователю не нужно знать, где находится их контент и им редко приходится думать о том, какую версию контента они просматривают сейчас. Следующие рекомендации помогут вам обеспечить это в вашем приложении.

Уважайте пользовательские аккаунты iCloud. Важно помнить, что хранилище iCloud является ограниченным ресурсом, за который пользователи платят. Вы должны использовать хранилище iCloud для хранения информации, которую пользователи могут создавать и понимать, но избегать его использования для хранения ресурсов, которые можно восстановить. Кроме того отметим, что когда пользовательский аккаунт iCloud активен, iCloud автоматически создает резервные копии содержимого документов и папок приложения. Чтобы избежать использования большого пространства пользователем, будьте придирчивы к содержанию, помещенному в папку Documents.

Определите типы информации для хранения в хранилище iCloud. В дополнение к документам и другому контенту пользователя, можно также хранить небольшое количество ключей и значений данных в хранилище iCloud. Например, если ваше приложение помогает пользователям просматривать контент (книги или журналы), вы можете использовать хранилище iCloud для хранения последней просмотренной страницы, таким образом, когда пользователи откроют контент на другом устройстве, они могут продолжить с того места, где остановились.

Если вы храните предпочтения в хранилище iCloud key-value, убедитесь, что предпочтения пользователей будут доступны на всех их устройствах. Например, некоторые предпочтения более полезны в рабочей среде, чем в домашних условиях, поэтому име-

ет смысл хранить предпочтения на сервере вашего приложения, а не в пользовательском аккаунте iCloud, чтобы их настройки находились в доступности, вне зависимости от доступности iCloud.

Убедитесь, что ваше приложение ведет себя разумно, когда хранилище iCloud не доступно. Например, если пользователь вышел из своего аккаунта iCloud, выключите использование iCloud для вашего приложения или включите Airplane mode, пока хранилище iCloud недоступно. В этом случае пользователи выполняли действия по выключению доступа к хранилищу iCloud, так что вашему приложению не требуется сообщать им об этом. Тем не менее, может быть разумным показать пользователям, что они создают изменения, которые не будут видны на других устройствах, до тех пор, пока они не восстановят доступ к хранилищу iCloud.

В случае необходимости делайте простым для пользователей включение хранилища iCloud в вашем приложении. На своих iOS-устройствах, пользователи входят в аккаунт в настройках iCloud и ожидают, что их приложения будут автоматически подключаться к iCloud. Но если считаете, что пользователь должен выбрать, использовать iCloud или нет, вы можете предоставить простой вариант, который они могут установить при запуске вашего приложения. В большинстве случаев эта опция должна обеспечить выбор между использованием хранилища iCloud для всего содержимого, к которому пользователи получают доступ в вашем приложении или нет.

Избегайте выбора пользователем документов, сохраняющихся на iCloud. Как правило, пользователи ожидают, что все интересующее их содержание будет доступно через хранилище iCloud. Ваше приложение не должно требовать от пользователей добавления в хранилище файлов на поэтапной основе. Чтобы обеспечить надлежащее взаимодействие с пользователем, возможно, придется перестроить путь, с помощью которого ваше приложение обрабатывает и предоставляет содержание так, чтобы выполнять больше файлоуправляющих задач для пользователей.

Не давайте пользователям возможность создания “локальных” документов. Не зависимо от того, поддерживает ваше приложение хранилище iCloud или нет, вы не должны позволять пользователю мыслить категориями файловой системы конкретного устройства. Вместо этого вы должны сосредоточиться на повсеместной доступности их содержимого через iCloud.

При необходимости обновляйте содержимое автоматически. Лучше всего, когда пользователи не предпринимают никаких действий для получения доступа к самым последним версиям содержимого вашего приложения. Тем не менее, необходимо сбалансировать это знание с уважением пространства пользовательского устройства и ограничением полосы прокрутки. Если ваши пользователи работают с очень большими документами, возможно, будет логичным предоставить им контроль над загрузкой обновлений из хранилища iCloud. Если вам нужно сделать это, спроектируйте способ идентификации более поздних версий документа, доступных для скачивания. Когда пользователь выбирает обновление документа, необходимо обеспечить тонкую обратную связь, если загрузка занимает всего несколько секунд.

Предупреждайте пользователей о последствиях удаления документа. Когда пользователь удаляет документ в приложении, которое использует хранилище iCloud, документ будет удален из пользовательского аккаунта iCloud и со всех остальных устройств. Это подходит для отображения предупреждений, которые описывают этот результат для получения подтверждения, прежде чем выполнить удаление.

Сообщайте пользователям о конфликтах, когда это возможно, но только при необходимости. Используя программный интерфейс хранилища iCloud, вы должны быть

готовы разрешить большинство конфликтов между различными версиями документов без участия пользователя. В случаях, когда это невозможно, убедитесь, что вы обнаружите конфликты как можно скорее, таким образом, помогая пользователю избежать потери времени на неправильную версию их содержания. Необходимо разработать ненавязчивую форму, чтобы показать пользователю наличие конфликта, а затем чтобы пользователи могли легко различать варианты и принимать решения.

Не забудьте включить пользовательский iCloud-контент в поиск. Пользователи с аккаунтами iCloud склонны считать, что их контент будет везде доступен, и они ожидают, что результаты поиска будут это учитывать. Если ваше приложение позволяет пользователям искать свой контент, убедитесь, что вы используете соответствующие API для расширения поиска на их iCloud-аккаунты. (чтобы узнать, как осуществлять поиск по контенту в хранилище iCloud, см. “Incorporating Search into Your Infrastructure” в iOS App Programming Guide .)

Multitasking (Многозадачность)

Многозадачность позволяет людям быстро переключаться между недавно использовавшимися приложениями, поскольку приложения могут быть свернуты в фоновый режим, когда пользователи выходят из них. Приостановленное приложение может возобновить свою работу быстро, поскольку не требуется заново загружать его UI.

Пользователи используют многозадачную панель (как показано здесь – внизу iPhone-приложения Календарь), чтобы выбрать недавно использовавшиеся приложения.



Успешность использования многозадачной среды осуществляется путем достижения гармоничного сосуществования с другими приложениями на устройстве.

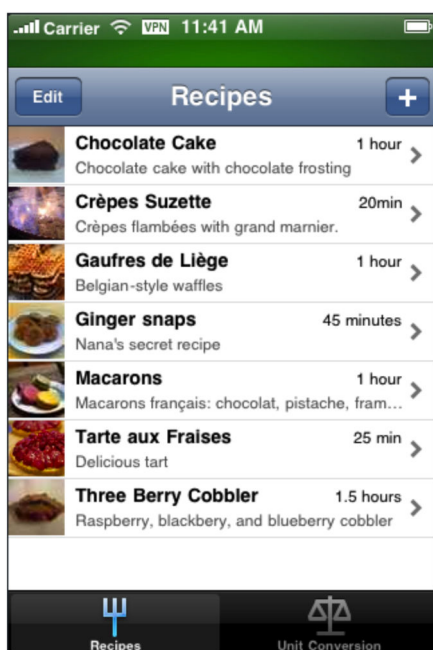
На более высоком уровне это означает, что все приложения должны:

- Изящно управлять прерыванием работы и аудио потоками других приложений;
- Останавливать и перезапускать (то есть переходить к приложению из фонового режима) быстро и плавно;
- Корректно продолжать работать в свернутом состоянии.

Следующие точные рекомендации помогут вашему приложению преуспеть в многозадачной среде.

Будьте готовы к сбоям и возобновлениям работы. Многозадачность увеличивает вероятность того, что фоновое приложение прервет работу вашего приложения. Другие функции, такие как наличие рекламы и быстрое переключение приложений, могут стать причиной более частых сбоев. Чем быстрее и точнее вы сможете сохранить текущее состояние вашего приложения, тем быстрее пользователи смогут возобновить или продолжить работу с момента возникновения сбоя.

Убедитесь, что ваш UI обладает double-high status bar. Double-high status bar появляется во время таких событий как телефонные звонки, запись звука, а также набор номера. В неподготовленном приложении дополнительная высота этой панели может вызвать проблемы разметки. Например, UI может сворачиваться или закрываться. В многозадачной среде особенно важно быть готовым справиться с Double-high status bar правильно, поскольку есть вероятность того, что будет больше приложений, вызывающих его появление. Вы можете вызвать double-high status bar во время тестирования, чтобы проще было найти и исправить любые окна, которые не обрабатывают его должным образом. (Чтобы узнать, как сделать это с помощью iOS Simulator см. "Manipulating the Hardware" в Tools Workflow Guide for iOS.)



Будьте готовы приостановить деятельность, которая требует внимания пользователей и их активного участия. Например, если ваше приложение является игровым или служит для просмотра медиа, убедитесь, что ваши пользователи не пропустят никакой контент или событие, покидая ваше приложение. Когда пользователи возвращаются к игре или приложению для просмотра медиа, они хотят продолжить работу с ним, будто они и не покидали его.

Убедитесь, что аудио ведет себя должным образом. Многозадачность делает более вероятным другую медиа-активность, во время работы вашего приложения. Это также повышает вероятность того, что аудио придется приостановиться, а затем возобновиться для обработки прерываний. Для получения точных рекомендаций, которые помогут вам убедиться, что аудио отвечает ожиданиям пользователей и правильно сосуществует с другими аудио-устройствами, см. раздел "Sound" (стр. 84).

Экономичное использование локальных уведомлений. Приложение может организовывать локальные уведомления, отправляющиеся в определенные моменты: если приложение приостанавливается, работает в фоновом режиме, или не работает вообще. Для хорошего пользовательского восприятия, избегайте утомления пользователей слишком большим количеством уведомлений, а также следуйте указаниям по созданию содержания уведомлений, описанным в разделе "Notification Center" (стр. 72).

При необходимости завершайте инициированные пользователем задачи в фоновом режиме. Когда пользователи инициируют задачи, они обычно ожидают завершения его работы, даже если они выходят из вашего приложения. Если ваше приложение находится в процессе выполнения инициированной пользователем задачи, которая не требует дополнительного взаимодействия с пользователем, вы должны завершить его в фоновом режиме перед отключением.

Notification Center (Центр Уведомлений)

Центр Уведомлений предоставляет пользователям единое удобное место для просмотра уведомлений от приложений. Пользователи ценят ненавязчивый интерфейс Центра Уведомлений и возможность индивидуализировать способ предоставления уведомлений для каждого приложения.

Центр Уведомлений использует секционный список для отображения новых объектов уведомлений от приложений, которые интересуют пользователей, в дополнение к уведомлениям, которые пользователи могут также выбрать для просмотра погоды и информации фондового рынка в Центре Уведомлений.



iOS-приложения могут использовать локальные или Push-уведомления, чтобы известить пользователей об интересных событиях, таких как:

- Получение сообщения;
- Напоминание о событии;
- Наличие доступных обновлений;
- Статус каких-либо изменений.

Локальные уведомления запланированы приложением и предоставлены iOS на одном устройстве, не зависимо от состояния приложения (в фоне или развернуто). Например, Календарь или Список Дел приложения могут назначать местные уведомления, чтобы предупредить пользователей о предстоящей встрече или событии.

Push-уведомления отправляются удаленным сервером приложения на Apple Push Notification сервис, который передает уведомления на все устройства, на которых установлено приложение. Например, игра, в которую пользователь может играть против удаленных противников, может обновить всех игроков за последний шаг.

Вы по-прежнему получаете локальные и Push-уведомления, когда ваше приложение работает на переднем плане, но передаете информацию пользователям специфическим образом внутри приложения.

iOS-приложения, поддерживающие локальные или Push-уведомления, могут участвовать в Центре Уведомлений по-разному, в зависимости от пользовательских предпочтений. Чтобы убедиться, что пользователи могут индивидуализировать свой опыт уведомлений, вы должны поддерживать как можно больше из следующих способов оповещения:

- Баннер (Banner);
- Предупреждение (Alert);
- Бэдж (Badge);
- Звук (Sound).

Баннер – это небольшое полупрозрачное окно, которое появляется на экране, а затем исчезает через несколько секунд. В дополнение к вашему уведомлению iOS отображает уменьшенную версию иконки вашего приложения в баннере, чтобы пользователи могли сразу увидеть, какое приложение уведомляет (чтобы узнать больше о маленьких иконках приложений, см. “Small Icons” (page 171)).



Предупреждение является стандартным окном предупреждений, которое отображается на экране и требует взаимодействия с пользователем для освобождения. Вы предоставляете сообщение уведомления и, возможно, название кнопки действия в предупреждении. Вы не контролируете фон и отображение предупреждения или кнопок.



Бэйдж предоставляет собой маленький красный овал, который отображает количество ожидающих элементов уведомлений (бэйдж отображается в правом верхнем углу иконки приложения). У вас нет контроля над размером и цветом бэйджа.



Индивидуализированные или предоставляемые системой звуки могут сопровождать любые из трех описанных выше вариантов оповещения.

-----**Примечание:** Приложения, которые используют локальные уведомления, могут предоставить баннеры, предупреждения, бэйджи и звуки. Но приложения, использующие Push-уведомления вместо локальных, могут обеспечить только те типы уведомлений, которые соответствуют Push-категориям, которые зарегистрированы приложением. Например, если Push-уведомления приложения регистрируют только предупреждения, пользователям не предоставляется выбор получить бэйджи и звуки при получении уведомления.

При разработке поставляемого уведомлениями контента, обязательно соблюдайте следующие правила.

Держите содержание бэйджа в актуальном состоянии. Это особенно важно при обновлении бэйджа как только пользователь получает новую информацию, таким образом, чтобы они не думали, что они получили дополнительные уведомления. Отметим, что установление значения бэйджа в ноль влечет за собой уведомление соответствующих пунктов из Центра Уведомлений.

Не отправляйте несколько уведомлений об одном и том же событии. Пользователи могут посещать пункты уведомлений по выбору; элементы не исчезают, пока пользователи не обработают их неким образом. Если вы отправляете несколько уведомлений об одном и том же событии, вы пополняете список Центра Уведомлений, и пользователи могут отключить уведомления от вашего приложения.

Предоставляйте уникальный текст, не содержащий название вашего приложения. Ваше уникальное сообщение отображается в предупреждениях и баннерах, в списке пунктов Центра Уведомлений. Вы не должны включать название вашего приложения в уникальное сообщение, поскольку iOS автоматически отображает название рядом с вашим сообщением.

Для большего удобства текст локальных и Push-уведомлений должен:

- Сосредотачивать на информации, а не на действиях пользователя. Избегайте сообщений, типа “кнопка предупреждения для нажатия” или как открыть ваше приложение.
- Будьте достаточно лаконичными, чтобы отобразить сообщение в одной или двух строках. Длинные сообщения являются трудными для быстрого чтения пользователями, и они могут заставить предупреждение прокрутиться.
- Используйте предложения капитализационного стиля с соответствующими знаками препинания. По возможности, используйте законченные предложения.

Примечание: В основном элементы Центра Уведомлений могут отображать больше уведомлений, чем баннер. При необходимости iOS обрезает сообщение так, чтобы оно хорошо вписывалось в стиль каждого уведомления о доставке, и для достижения наилучших результатов вы не должны сокращать свои сообщения.

При желании предоставляйте уникальный заголовок для кнопки действия в предупреждении. Предупреждение может содержать одну или две кнопки. В двух кнопочном баннере кнопка Закрывать слева, а кнопка под названием View (по умолчанию) – справа. Если вы укажете одну кнопку, предупреждение отобразит кнопку ОК.

Нажатие на кнопку действия освобождает от предупреждения и одновременно запускает приложение. При нажатии любой кнопки “Закрывать” или кнопки “ОК” отклоняется предупреждение, а приложение не запускается.

Если вы хотите использовать уникальные названия кнопок действия, не забудьте создать заголовок, который четко описывает действие, происходящее при запуске приложения. Например, игра может использовать название Play, показывая, что нажатие кнопки открывает приложение в том месте, где пользователь может занять очередь. Убедитесь, что название:

- Использует название в капитализационном стиле;
 - Является достаточно коротким, чтобы поместиться в кнопке без усечения (не забудьте также проверить длину локализованного названия).
-

Примечание: Ваши уникальные названия кнопок, должны быть также отображены в “slide to view” сообщениях, когда приходит уведомление, во время блокировки устройства. При этом ваши уникальные названия автоматически преобразуются в нижний регистр и заменяются словом “view” в сообщении.

Предоставляйте звуки, которые пользователи могут выбрать для прослушивания, в момент получения уведомления. Звук может привлечь внимание пользователей, когда они не смотрят на экран устройства. Пользователи, возможно, захотят включить звук, когда они ожидают важное уведомление. Например, приложение Календарь может воспроизводить звук с предупреждением, чтобы напомнить людям о неизбежном событии. Или приложение управления совместными задачами (collaborative task management) может воспроизводить звук с обновлением бэйджа, чтобы сигнализировать о том, что удаленный коллега завершил задание.

Вы можете задавать уникальные звуки или использовать встроенный звуковой сигнал. Если вы создаете уникальный звук, убедитесь, что он короткий, уникальный и профессиональный. (Чтобы узнать о технических требованиях к этим звукам, см. “Preparing Custom Alert Sounds” в Local and Push Notification Programming Guide.) Обратите внимание, что вы не можете вызвать программно вибрацию устройства при предоставлении уведомления, так как пользователь может управлять вибрацией, сопровождающей предупреждения.

При необходимости предоставляйте изображение запуска. В дополнение к отображению ваших существующих изображений запуска вы можете указать другое изображение для отображения, когда пользователи запускают ваше приложение в ответ на уведомление. Например, игра может указать запуск изображения, которое похоже на экран

в игре, в отличие от изображения, похожего на экран стартового меню. Если вы не предоставляете изображение запуска, iOS отображает либо предыдущий скриншот, либо одно из прочих изображений запуска. (Чтобы узнать, как создать изображение запуска, см. “Launch Images” (стр. 163).)

Printing (Печать)

В iOS 4.2 и выше, а так же на устройствах, поддерживающих многозадачность, пользователи могут использовать беспроводную печать контента приложения. Вы можете воспользоваться встроенной поддержкой печати изображений и PDF-файлов или использовать печать с помощью конкретных программ для уникального форматирования и рендеринга. iOS управляет открытием принтера, планированием и выполнением задач на печать выбранного принтера.

Как правило, пользователи нажимают стандартную кнопку действия в вашем приложении, когда хотят напечатать что-либо. Когда они выбирают Print-элемент в представленном окне, то могут выбрать принтер, установить доступные параметры печати и нажать кнопку Print, чтобы начать процесс. На iPhone это окно отображается в списке действий (action sheet), которое скользит вверх от нижней части экрана вверх, на iPad – отображается всплывающим окном, которое возникает от кнопки.



Пользователи могут проверить состояние процесса, заданного ими в Центре Печати, и являющегося фоновым системным приложением, которое доступно только во время выполнения процесса печати. В Центре Печати пользователи могут просматривать текущие очереди печати, получать подробную информацию о конкретном процессе и даже отменить задачу.

Вы можете поддерживать основную печать в вашем приложении, используя сравнительно немного дополнительного кода (для получения сведений о добавлении поддержки печати в коде, см. Drawing and Printing Guide for iOS). Для того чтобы пользователи оценили опыт печати в вашем приложении, следуйте следующим рекомендациям:

Используйте предоставляемую системой кнопку действия. Пользователи уже знакомы со смыслом и поведением этой кнопки, поэтому хорошо использовать ее, по возможности. Основным исключением является случай, когда ваше приложение не содержит панели инструментов или панели навигации. Тогда вам нужно создать уникальную кнопку печати, которая может появиться в основном пользовательском интерфейсе вашего приложения, потому что кнопка действия может быть использована только в панели инструментов или панели навигации.

Отображайте пункт Печать, когда печать является одной из основных функций текущего контекста. Если печать неуместна в данном контексте, или, если пользователи, вероятно, не хотят печатать, не включайте элемент Печать в отображаемом окне кнопки действия.

В случае необходимости предоставляйте дополнительные возможности печати пользователям. Например, вы могли бы предложить пользователям выбрать диапазон страниц или запросить несколько копий.



Не отображайте print-specific UI, если пользователь не может печатать. Убедитесь в том, что вы проверили устройство пользователя на поддержку печати до того, как отображали пользовательский интерфейс, который предлагает печать в качестве опции. Чтобы узнать, как сделать это в коде, см. [UIPrintInteractionController Class Reference](#).

iAd Rich Media Ads (iAd Мультимедийные Объявления)

iOS 4.0 и более поздние версии поддерживают рекламы на странице в пределах вашего приложения, и вы можете получить доход, когда пользователи будут видеть и взаимодействовать с ними. Важно, чтобы вы спланировали, когда и как интегрировать объявления в UI, чтобы пользователи были мотивированы на их просмотр, не отвлекаясь от вашего приложения.

Вы размещаете и поддерживаете отображения рекламы с помощью iAd Network в определенном окне вашего UI. Первоначально это окно содержит рекламный баннер, который функционирует как вход в полное рекламное объявление. Когда пользователи касаются баннера, реклама выполняет запрограммированные действия, такие как воспроизведение видео, отображение интерактивного содержимого или запуск Safari для открытия web-страницы. Действие может отображать содержимое, которое охватывает ваш UI, или это может привести к переходу вашего приложения на второй план.

Есть два вида баннеров, которые можно отобразить в приложении: стандартные и полноэкранные. Оба типа баннера служат одной цели — отобразить пользователю рекламу, но отличаются по внешнему виду и функциональности.

Стандартный баннер занимает небольшую площадь экрана и часто виден до тех пор, пока отображается экран. Вы можете выбрать окно приложения, в котором будет отображаться баннер, и освободить место для баннера в окне макета. Все iOS-приложения, работающие в iOS 4.0 и более поздних версиях, поддерживают отображение стандартного баннера. Вы используете окно, предоставляемое `ADBannerView` классом, чтобы включить стандартный баннер в ваше приложение.

Полноэкранный баннер занимает большую часть или весь экран и, как правило, отображается в определенное время работы приложения или в конкретных местах. Вы можете выбрать, следует ли отображать баннер модально или в виде отдельной страницы в пределах прокручиваемого содержимого.

Полноэкранные баннеры поддерживаются только приложениями iPad, запущенными на iOS 4.3 и более поздних версиях. Вы используете окно, предоставляемое `ADInterstitialAd` class, чтобы использовать полноэкранный баннер в вашем приложении.

Оба типа баннеров появляются внутри `iAd` frame, который отображает `iAd` значок в правом нижнем углу. `iAd` frame был разработан для улучшения отображения при прикреплении к нижнему краю вашего приложения.

Размеры стандартного баннера, помещенного в ваше приложение, могут варьироваться в зависимости от вращения устройства и его ориентации.

Таблица 6-1 Размеры стандартного баннера

Устройство	Вертикальная ориентация	Горизонтальная ориентация
iPad	768 x 66	1024 x 66
iPhone	320 x 50	480 x 32

Габаритные размеры полноэкранного баннера могут изменяться в зависимости от изменения ориентации устройства, но высота в дальнейшем изменяется при наличии или отсутствии iOS панелей (например, панели инструментов и панели ярлыков) в UI вашего приложения. Ширина полноэкранного баннера всегда равна ширине экрана iPad.

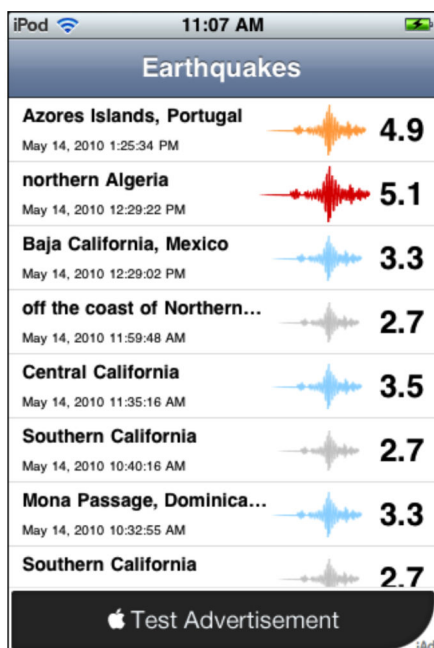
Таблица 6-2 Размеры полноэкранного баннера

Ориентация iPad	Высота	Ширина
Вертикальная ориентация	От 911 до 1024	768
Горизонтальная ориентация	От 655 до 768	1024

Чтобы обеспечить полную интеграцию баннеров и обеспечить максимальное удобство для пользователей, следуйте следующим основным принципам:

Располагайте стандартный баннер вблизи нижней части экрана. Такое расположение может несколько отличаться в зависимости от того, есть ли панель на нижней части экрана и если есть, какая именно.

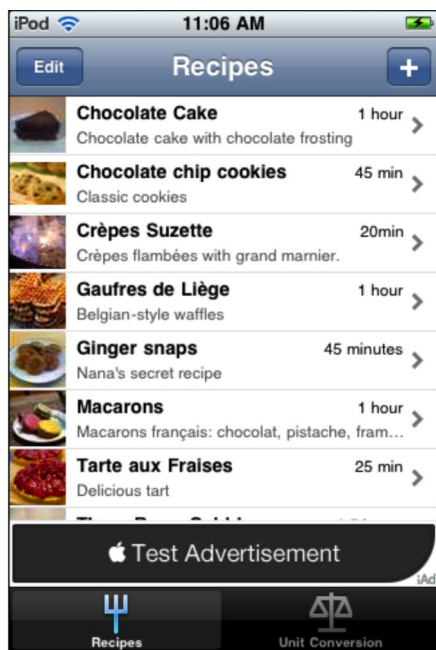
Если в нижней части экрана нет панелей, располагайте стандартный баннер у нижнего края экрана (как показано ниже).



Если там вообще нет панелей, также располагайте стандартный баннер у нижнего края экрана (как показано ниже).



Если присутствует панель инструментов или панель ярлыков, располагайте баннер сразу над панелью инструментов или панелью ярлыков (как показано ниже).

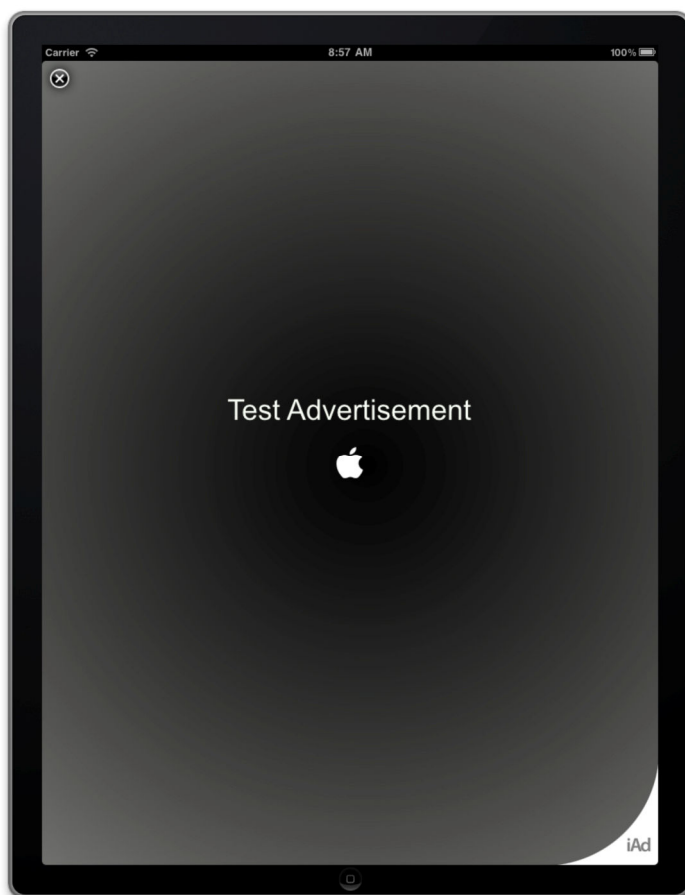


Предоставляйте полноэкранный баннер в тот момент, когда имеется пауза в пользовательском восприятии приложения (модально). Если есть естественные перерывы или изменения контекста в работе вашего iPad-приложения, модальный стиль презентации может быть целесообразным. Когда вы предоставляете модальный полноэкранный баннер (с помощью `presentFromViewController`), пользователь должен либо принять объявление, либо отклонить.

По этой причине отличной идеей будет использование модального стиля презентации, когда пользователи ожидают изменения в восприятии, например, при завершении задачи.

Будьте уверены, что открытие и закрытие модального полноэкранного баннера, имеет смысл в вашем приложении. Например, игра может использовать плавные анимации появления (fade-in) и исчезновения (fade-out) при раскрытии и закрытии баннера.

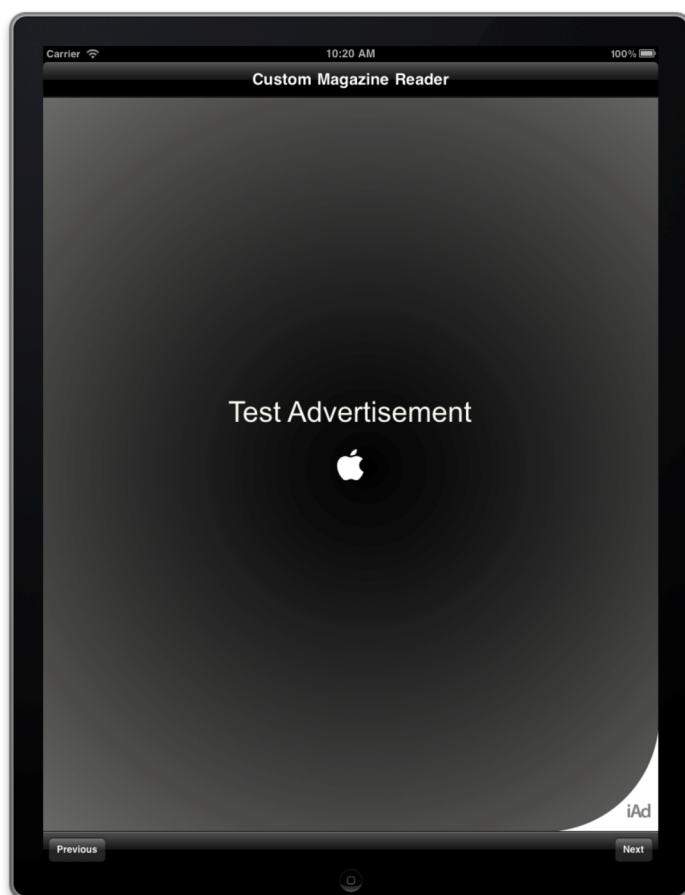
Как видите, ниже представлен модальный полноэкранный баннер, полностью закрывающий приложение и включающий в себя UI и iAd-кнопку закрытия в левом верхнем углу.



Предоставляйте немодальный полноэкранный баннер, когда существуют переходы между двумя окнами. Если пользовательский опыт вашего приложения включает в себя частые переходы по экранам, например, разбиение по страницам журнала или резкие передвижения по галерее, немодальный стиль презентации может быть целесообразным. Когда вы представляете немодальный полноэкранный баннер (с помощью `presentInView`), вы можете сохранить панели в вашем UI, чтобы пользователи могли использовать управление приложением и переместиться назад или вернуться к объявлению. Как и все баннеры, полноэкранный баннер запускает iAd при нажатии на него пользователем, но ваше приложение может реагировать и на другие жесты в области баннера (например, перетаскивание или перелистывание), если это требуется.

Обязательно используйте соответствующие анимации для появления и скрытия немодальных полноэкранных баннеров. Например, в приложении Magazine Reader присутствует баннер, использующий ту же анимацию, что и при переходе на другие страницы.

Немодальный полноэкранный баннер, показанный ниже, отображается между навигационной панелью и панелью инструментов. Однако вы сами выбираете, какие панели должны отображаться.



Убедитесь, что все баннеры появляются, когда и где это имеет смысл в вашем приложении. Пользователи чаще входят в iAd, когда они не чувствуют, что прерывают для этого рабочий процесс. Это особенно важно для предоставления в таких приложениях, как игры: вы не должны размещать окна баннера там, где они будут конфликтовать с игрой.

Избегайте отображения баннеров на экранах, которые пользователи могут видеть только очень короткое время. Если ваше приложение содержит экраны, по которым пользователи быстро перемещаются, углубляясь и переходя к содержанию, лучше позаботиться о том, чтобы баннеры не отображались на этих экранах. Пользователи скорее заметят баннер, который остается на экране более 2 секунд.

По возможности показывайте баннеры для двух ориентаций. Лучше всего, когда пользователям не приходится менять ориентацию устройства для переключения между использованием вашего приложения и просмотром реклам. Кроме того, поддержка обеих ориентаций позволяет использовать более широкий спектр рекламных объявлений. Чтобы узнать, как баннер реагирует на изменение ориентации, смотрите "iAd Programming Guide".

Не допускайте прокрутки стандартный баннер на экране. Если ваше приложение предусматривает прокрутку содержимого экрана, убедитесь, что стандартный баннер остается на предусмотренном для него месте.

Когда пользователи просматривают рекламу и взаимодействуют с ней, приостанавливайте активность, которая требует их внимания и взаимодействия. Когда пользователи выбирают просмотр объявлений, они не хотят чувствовать себя так, будто они пропускают события в вашем приложении, и не хотят, чтобы ваше приложение

было прервано объявлением. Правилom хорошего тона считается приостановка действий приложения так, словно оно ушло в фоновый режим.

Не останавливайте рекламу, за исключением некоторых случаев. В общем, ваше приложение продолжает работать и получать события, в то время как пользователи просматривают и взаимодействуют с рекламой, так что вполне возможно, что произойдет событие, настоятельно требующее их немедленного внимания. Тем не менее, очень мало сценариев, требующих закрытия просматриваемого объявления. Одна из возможностей заключается в приложении, которое предоставляет передачу голоса по протоколу IP (VoIP) службы. В таких приложениях, вероятно, имеет смысл отменить работу объявления при поступлении входящего звонка.

Примечание: Отмена объявления может отрицательно сказаться на впечатлении от рекламы, получаемой вашим приложением и доходе, который вы можете получить.

Quick Look Document Preview (Быстрый Предпросмотр Документа)

На iOS 4 и более поздних версиях пользователи могут просматривать миниатюры документов в рамках вашего приложения, даже если ваше приложение не позволяет открытие этих документов. Например, можно разрешить пользователям просмотр документов, загруженных из интернета или взятых из других источников. Что бы узнать больше о поддержке Quick Look Document Preview в вашем приложении, посмотрите "Document Interaction Programming Topics" для iOS.

Прежде чем пользователи увидят превью документа в вашем приложении, они могут видеть информацию о документе в индивидуализированном окне, созданном вами. Например, после того, как пользователи загружает документ, прилагаемый к сообщению электронного письма, Mail на iPad отображает иконку документа, его название и размер в индивидуализированном окне сообщения. Пользователи должны нажать на это окно, чтобы перейти к предварительному просмотру документа.



Вы можете преподнести предварительный просмотр документа в новом окне вашего приложения либо в полноэкранном, модальном окне. Способ преподнесения вы выбираете в зависимости от того, с каким устройством работает ваше приложение.

На iPad предварительный просмотр документа модален. Большой экран iPad подходит для отображения предварительного просмотра документа, давая понять пользователю, что он может выйти. Zoom-переход особенно подходит для отображения предварительного просмотра.

На iPhone предварительный просмотр документа отображается в отдельном окне, желательно в окне навигации. Это позволит пользователям перемещаться из предварительного просмотра документа без потерь контекста в вашем приложении. Модальное отображение предварительного просмотра документа на iPhone возможно, но не рекомендовано. (Заметим, что zoom-переход недоступен на iPhone).

Кроме того, обратите внимание, что отображение предварительного просмотра документа в экране навигации Quick Look позволяет разместить предварительный просмотр специальными контроллерами навигации на панели навигации. (Если ваше окно уже содержит панель инструментов, лучше вместо нее разместите Quick Look контроллеры навигации предварительного просмотра.)

Sound (Звук)

iOS-устройства производят звук отличного качества, что пользователи ценят. В вашем приложении звук может быть важной частью пользовательского восприятия или обеспечивать только второстепенное расширение восприятия. Независимо от того, какую роль играет звук в вашем приложении, вы должны знать, что пользователи от него ожидают, и он должен оправдывать их ожидания.

Понимайте Пользовательские Ожидания

Пользователи могут использовать устройство для регулирования звука, и они могут использовать проводные или беспроводные гарнитуру и наушники. Также у пользователей есть различный набор способов воздействия на звук, который они слышат. Хотя вы можете найти некоторые из действий удивительными, все они следуют принципу пользовательских элементов управления в том, что пользователь, а не устройство, решает, когда уместно включить звук, а когда нет.

Пользователи переключаются в беззвучный режим, когда им требуется:

- Не отвлекаться на неожиданные звуки, такие как телефонные рингтоны или звуки входящих сообщений.
- Не слышать звуки, которые являются результатом действия пользователя, например клавиатура или другие звуки обратной связи, случайные звуки или звуки запуска приложения.
- Избегать слышимых звуков игры, которые не являются необходимыми для использования игры, например, случайные звуки и музыка.

Обратите внимание: люди переключают свои устройства в беззвучный режим, используя либо Ring/Silent переключатель (на iPhone), либо выключатель Silent (на iPad).

Например, в театрах пользователи переключают свои устройства в бесшумный режим, чтобы не беспокоить других людей. В данной ситуации пользователи по-прежнему хотят иметь возможность использовать приложения на своих устройствах, но они не хотят смущаться из-за звуков, которых они не ожидают, например, мелодий или звуков новых сообщений.

Ring / Silent (или переключатель Без Звука) переключатель не приглушает звуки, возникающие в результате действий пользователя, которые четко предназначены для создания звука. Например:

- Звук не будет отключен в приложениях, которые специально созданы для проигрывания медиа-файлов, потому что воспроизведение было явно и четко запущено пользователем.
- Будильник не замолчит, поскольку его работа была запрограммирована пользователем.
- Саунд-клип приложения Language-learning не замолчит, потому что пользователь принял действия, чтобы услышать это.
- Разговор в приложении Аудио-чат не затихнет, поскольку пользователь запустил приложение с целью аудио-общения.

Пользователи используют кнопки громкости для регулировки уровня громкости всех звуков, которые устройство может воспроизвести, включая песни, звуки приложения и звуки устройства. Пользователи могут использовать кнопки регулировки громкости, чтобы приглушить любой звук независимо от положения Ring / Silent (или переключателя Без Звука). Использование кнопки регулировки громкости для настройки приложения во время проигрывания аудио также регулирует общую громкость системы, за исключением громкости звонка.

iPhone: кнопка регулировки громкости, выступает в роли регулятора громкости звонка, пока аудио не воспроизводится.

Пользователи используют гарнитуры и наушники, чтобы слышать звуки в частном порядке и для освобождения рук. Независимо от того, являются ли эти аксессуары проводными или беспроводными, пользователи имеют определенные ожидания в их отношении.

Когда пользователи подключают гарнитуру или наушники или подключают беспроводной аудиоустройство, они намерены продолжать прослушивание текущего аудио, но персонально. По этой причине они ожидают, что приложение, воспроизводящее аудио в данный момент, продолжит свою работу без остановок.

Когда пользователи отключают гарнитуру или наушники или отсоединяются от беспроводного аудиоустройства (или устройство выходит из диапазона или выключается), они не хотят автоматически делиться с окружающими тем, что они слушали. По этой причине они ожидают, что приложение, которое в настоящее время воспроизводит аудио, встанет на паузу, позволяя им возобновить воспроизведение, когда они будут к этому готовы.

Определите Аудиоповедение Вашего Приложения

В случае необходимости можно настроить относительно независимые уровни громкости, чтобы создать наилучшее сочетание выхода аудио для вашего приложения. Но громкость окончательного вывода аудио всегда должна регулироваться системным регулятором, была ли она установлена кнопками регулировки громкости или слайдером громкости. Это означает, что контроль над аудиоприложениями остается в руках пользователя там, где это необходимо.

Убедитесь, что ваше приложение может отображать пикеры аудиомаршрута в случае необходимости.

(аудиомагистраль для звуковых сигналов, например, от устройства к наушникам или от устройства к колонкам). Даже если пользователи физически не подключают (отключают) беспроводные аудиоустройства, они все еще хотят иметь возможность выбора иного аудиомаршрута. Чтобы исправить это, iOS автоматически отображает элемент управления, который позволяет пользователям выбрать выходной аудиопоток (используйте класс `MPVolumeView`, чтобы отобразить элемент управления в вашем приложении). Иницилируемые пользователем действия выбора иного маршрута звука подразумевают для пользователей, что воспроизводимое в настоящее время аудио продолжится без остановки.

Если вам надо отобразить слайдер громкости, не забудьте использовать предоставляемый системой слайдер громкости, доступный при использовании `MPVolumeView` class. Заметим, что при активном выводе звука через устройство не поддерживается регулятор громкости, а слайдер громкости заменяется соответствующим именем устройства.

Если ваше приложение воспроизводит только звуковые эффекты интерфейса, связанные с его функциональностью, используйте Сервис Звуковой Системы. Сервисы Звуковой Системы технологии iOS, которые производят звуки предупреждений и UI, а также осуществляют вибрацию, не подходят для каких-либо других целей. При использовании сервисов звуковой системы для воспроизведения звука, вы не сможете контролировать взаимодействие аудио с аудио на устройстве, или его реагирование на прерывания и изменения в конфигурации устройства. Как пример проекта использования данных технологий см. "Audio UI Sounds (SysSound)".

Если звук играет важную роль в вашем приложении, используйте Audio Session Services или AVAudioSession class. Эти программные интерфейсы не производят звук, вместо этого они помогут вам выразить, как аудио должно взаимодействовать с аудиоустройством и реагировать на прерывания и изменения в конфигурации устройства.

iPhone: Независимо от того, какую технологию вы используете для создания аудио, или как вы определяете свой образ действий, телефон всегда может прервать работу приложения. Поэтому приложение не должно мешать пользователям получать входящие звонки.

В Audio Session Services функции аудиосессии выступают в качестве посредника для аудио между приложением и системой. Одним из наиболее важных аспектов аудиосессии является категория, которая определяет поведение аудио приложения.

Для того, чтобы реализовать преимущества служб аудиосеансов и обеспечить пользователей ожидаемыми аудиовпечатлениями, вам следует выбрать категорию, которая наиболее точно опишет аудиоповедение приложения. Это касается того, может ли ваше приложение воспроизводить звуки только на переднем плане или может также воспроизводить их и в фоновом режиме. Следуйте этим рекомендациям, как сделать этот выбор:

- **Выберите сессию аудиокатегорий на основе ее семантического значения, а не ее точного множества вариантов поведения.** Выбрав категорию, цель которой ясна,

можно быть уверенным, что ваше приложение ведет себя в соответствии с ожиданиями пользователей. Кроме того это даст больше шансов вашему приложению работать должным образом, если точный набор поведений уточнен на будущее.

● **В редких случаях добавляйте свойство аудиосессии изменения стандартного поведения категории.** Стандартное поведение категории осуществляет то, что большинство пользователей ожидают, поэтому вы должны тщательно подумать, прежде чем изменять это поведение. Например, вы можете добавить внутренних функций, для того чтобы убедиться, что ваше аудио громче, чем все другие (кроме аудио звонка), если это именно то, чего пользователи ожидают от вашего приложения. (Чтобы узнать больше о свойствах аудиосессии, см. "Fine-Tuning the Category" в "Audio Session Programming Guide".)

● **Рассмотрите, на чем базируется ваш выбор категории в текущей аудиосреде устройства.** Это может иметь смысл, если, например, пользователи могут использовать ваше приложение, слушая другое аудио, а не ваш саундтрек. Если вы делаете это, убедитесь, что не заставляете пользователей прекращать прослушивание их музыки или делать явный выбор саундтрека при запуске приложения.

● **В общем, избегайте изменения категории во время работы вашего приложения.** Основная причина для изменения категории - если ваше приложение должно поддерживать запись и воспроизведение в разное время. В данном случае лучше переключаться между категориями Record и Playback, по мере необходимости для выбора Play или Record категорий. Это потому, что выбор Record-категории гарантирует, что никакие предупреждения (например, оповещение о входящем текстовом сообщении) не будет звучать, пока идет запись.

Таблица 6-3 – списки аудиокатегорий сессий, которые вы можете использовать. Различные категории позволяют подавлять некоторые звуки Ring/Silent или Silent switch (или блокировка устройства), воспроизводить параллельно или воспроизводить во время того, как приложение находится в фоновом режиме. (Для выбранной категории и имен свойств и как они добавляются в программируемый интерфейс см. "Audio Session Programming Guide".)

Таблица 6-3 Категории аудиосессий и связанные с ними поведения

Категория	Значение	Приглушен- ный	Смешанный	Фоно- вый
Индивиду- альное ок- ружение	Звуки повышают функциональ- ность приложения, остальные звуки не воспроизводятся	Да	Нет	Нет
Окружение	Звуки повышают функциональ- ность приложения, но не за- ставляют замолкать другие	Да	Да	Нет
Категория	Значение	Приглушенный	Смешанный	Фоновый
Воспроизве- дение	Звуки имеют важное значение для функциональности прило- жения и могут смешиваться с другими	Нет	Нет (по умолча- нию) Да (когда смеши- вается с другими добавленными)	Да
Запись	Аудио, записанное пользовате- лем	Нет	Нет	Да
Воспроизве- дение и За- пись	Звуки предоставляют вход и выход, последовательно и од- новременно	Нет	Нет (по умолча- нию) Да (когда смешиваются с	Да

			другими)	
Аудиообработка	Приложение выполняет аппаратную поддержку кодирования аудио (не воспроизведение или запись)	N/A	Нет	Да*

* Если вы выбираете категорию аудиообработки и хотите выполнить обработку в фоновом режиме, то необходимо, чтобы ваше приложение не допускало его возврат из фона, до того как закончится обработка. Чтобы узнать, как это сделать, см. "Executing Code in the Background" в "iOS Application Programming Guide".

Вот несколько сценариев, которые иллюстрируют, как выбрать аудиокатегорию сессии, которая обеспечивает звучание, желаемое пользователями.

Сценарий 1: образовательное приложение, которое помогает людям учиться новому языку. Вы предоставляете:

- Обратную связь звуков, которые воспроизводятся, если пользователь касается определенных элементов управления
- Запись слов и фраз, которые воспроизводятся, когда пользователи хотят услышать примеры правильного произношения

В данном случае звук имеет важное значение для первичной функциональности. Пользователи используют это приложение, чтобы услышать слова и фразы на языке, которому они учатся, так что звук должен воспроизводиться даже тогда, когда устройство заблокировано или включен беззвучный режим работы. Поскольку пользователи должны ясно слышать звуки, они ожидают, что смогут остановить воспроизведение других аудио.

Пользователи рассчитывают на это приложение для получения аудиоопыта, и вам требуется использовать категорию Playback. Хотя эта категория может быть уточнена для параллельного воспроизведения с другими аудио, приложение должно использовать поведение по умолчанию, чтобы гарантировать, что другие аудио не конкурируют с образовательным контентом, выбранным пользователем.

Сценарий 2: передача голоса по протоколу IP (VoIP) приложения. Вы предоставляете:

- возможность принимать аудиовход;
- возможность воспроизведения аудио.

В этом приложении звук имеет важное значение для первичной функциональности. Зачастую пользователи используют это приложение, чтобы общаться с другими пользователями, хотя в настоящее время они пользуются другим приложением. Пользователи ожидают иметь возможность принимать звонки, когда они переключили свое устройство в беззвучный режим работы или устройство заблокировано, и они ожидают, что другое аудио не будет воспроизводиться на протяжении всего разговора. Они также ожидают иметь возможность получать и продолжать звонки, пока приложение находится в фоновом режиме.

Для получения ожидаемого пользовательского впечатления от этого приложения вы должны использовать воспроизведение и запись категории. Кроме того, вы должны быть уверены в том, что активация вашей аудиосессии производится только тогда, когда это нужно, чтобы пользователи могли использовать и другие аудио между вызовами.

Сценарий 3: Игра, позволяющая пользователям вести героя через ряд разных задач. Вы предоставляете:

- Различные звуковые эффекты, геймплей;
- Музыкальное сопровождение.

В этом приложении звук значительно повышает удобство работы, но не является необходимым для главной задачи. Кроме того, пользователи, скорее всего, оценят возможность играть в игру без звука или параллельно прослушивать песни своей музыкальной библиотеки, а не игровой саундтрек.

Лучшей стратегией будет узнать может ли запускаться ваше приложение во время воспроизведения других звуков. Не предоставляйте пользователям выбор между саундтреком и другим аудио. Вместо этого используйте Сервис Аудио Сессий функцию `AudioSessionGetProperty` для запроса состояния свойства `kAudioSessionProperty_OtherAudiolsPlaying`. На основании ответа вы сможете выбрать или Ambient категорию или Solo Ambient категорию (обе категории позволят пользователям играть в игру в тишине):

- Если пользователи слушают другие аудио, вы должны учесть, что они хотят продолжать слушать это аудио, вместо саундтрека игры.

В этой ситуации следует выбрать для вашего приложения категорию Ambient.

- Если пользователь не слушает никакой другой звук при запуске вашего приложения, в вашем приложении должна быть выбрана Solo Ambient категория.

Сценарий 4: приложение, которое обеспечивает точные, в режиме реального времени, навигационные инструкции для получения маршрута к месту назначения пользователя. Вы предоставляете:

- Разговорное руководство на каждом этапе путешествия;
- Несколько звуков обратной связи;
- Возможность продолжать слушать собственное аудио для пользователей.

В этом приложении голосовые инструкции навигации представляют основную задачу независимо от того, что приложение находится в фоновом режиме. По этой причине вы должны использовать Playback категорию, которая позволяет вашему аудио играть, когда устройство заблокировано или переключено на беззвучный режим, и в то время как приложение находится в фоновом режиме.

Чтобы дать людям возможность слушать другие аудио в то время как они используют ваше приложение, вы можете добавить `kAudioSessionProperty_OverrideCategoryMixWithOthers` property. Однако вы также должны убедиться, что пользователи могут слышать голосовые инструкции поверх звука, играющего в настоящее время. Для этого вы можете обратиться к `kAudioSessionProperty_OtherMixableAudioShouldDuck` свойству аудиосессии. Это гарантирует, что ваш звук громче, чем все, в настоящее время воспроизводимое, аудио (кроме аудио звонка на iPhone).

Сценарий 5: приложение блоггинга, позволяющее пользователям загружать их тексты и графику на web-сайт. Вы предоставляете:

- короткий звуковой файл запуска;
- Различные короткие звуковые эффекты, сопровождающие действия пользователя (например, звук, который играет, когда сообщение было загружено);
- звуковой сигнал, который играет при неудачном размещении.

В этом приложении звук расширяет возможности пользователя, но это случайность. Основная задача приложения никак не связана с аудио, и пользователям не нужно слышать звуки, для успешного использования приложения. В этом случае вы должны ис-

пользовать услуги звуковой системы для получения звука. Это происходит потому, что аудиоконтекст всех аудио в приложении соответствует назначению этой технологии, которая является результатом звуковых эффектов UI и предупреждающих звуков, которые подчиняются блокированию устройства и Ring / Silent (или Silent) переключателю так, как пользователи ожидают.

Управление Прерываниями Аудио

Иногда воспроизводимое в данный момент аудио прерывается аудио из различных приложений. Например, входящий звонок прерывает текущие аудиоприложения iPhone на время вызова. В многозадачной среде частота таких прерываний звука может быть высокой.

Для воплощения пользовательских ожиданий от аудио впечатлений, iOS основывается на следующем:

- Определение типа прерывания аудио, вызванного запуском вашего приложения;
- Соответствующая реакция при возобновлении работы приложения после прерывания аудио.

Каждое приложение должно определять аудиоперерывание, вызванное им, но не каждое приложение должно определять реакцию после окончания прерывания аудио. Это потому, что для большинства типов приложений адекватным ответом на завершение аудиоперерывания, будет возобновление воспроизведения аудио. Только приложения, которые полностью или частично являются приложениями воспроизведения медиа, а также обеспечивающие контроль воспроизведения медиафайлов, должны предпринимать соответствующие шаги для определения соответствующей реакции.

Концептуально есть два типа прерывания аудио в зависимости от типа аудио, инициирующего это прерывание, и от ожиданий пользователей касательно того, как аудиоприложение реагирует на завершение прерывания:

- **Возобновление прерывания.** Возобновляемые прерывания, инициируемые аудио, пользователи рассматривают как временную интерлудию в их первичном впечатлении от прослушивания.

После завершения возобновляемого прерывания приложение, отображающее элементы управления медиавоспроизведением, должно возобновлять выполнение задач, выполняемых до прерывания, будь то воспроизведение аудио или приостановка воспроизведения. Аудиоприложение, которое не имеет контроля воспроизведения медиафайлов, должно возобновлять воспроизведение аудио.

Рассмотрим пример: пользователь слушает приложение для воспроизведения музыки на iPhone и при получении вызова VoIP находится в середине песни. Отвечая на вызов, пользователь ожидает, что воспроизведение приостановится на время его разговора. По завершении вызова, пользователь ожидает, что приложение аудиовоспроизведения автоматически продолжит воспроизведение песни, потому как музыка, а не вызов, является основной функцией, и воспроизведение не было остановлено пользователем до вызова. С другой стороны, если пользователь приостановил воспроизведение аудио до вызова, он не ожидает автоматического включения воспроизведения аудио.

Другие примеры приложений, инициирующих возобновляемые прерывания, приложения сигнализации, аудио подсказки (например, подсказки направления движения), или других прерывистых звуков.

- **Не возобновляемые прерывания.** Не возобновляемые прерывания инициируются аудио, которые пользователи рассматривают как первичные звучания, например, аудио в при-

ложении воспроизведения мультимедиа. По окончании не возобновляемых перерывов, контроллеры приложения, которое отображает воспроизведение медиафайлов, не должны возобновлять воспроизведение аудио. Приложения, которые не имеют контроллеров воспроизведения медиафайлов, должны возобновлять воспроизведение аудио.

Рассмотрим пример: пользователь слушает приложение воспроизведения музыки (аудио приложение 1), когда другое приложение воспроизведения музыки (аудио приложение 2) прерывает. В ответ пользователь решает, слушать ли некоторое время музыку приложения 2. После выхода из воспроизведения приложения 2 пользователь не ожидает, что музыка приложения 1 автоматически начнет воспроизводиться, потому что воспроизведение аудио приложения 2 умышлено выбрано основным впечатлением от прослушивания.

Следующие рекомендации помогут вам решить, какую информацию предоставлять и как продолжать воспроизведение аудио по окончании прерываний.

Определите тип аудиоперерывания, вызванного вашим приложением. Это можно сделать путем отключения аудиосессии в одном из следующих двух способов, когда аудио завершилось:

- Если ваше приложение вызвало возобновляемое прерывание, отключите аудиосессию с помощью метки `AVAudioSessionSetActiveFlags_NotifyOthersOnDeactivation`.
- Если ваше приложение вызвало невозобновляемое прерывание, отключите ваши аудиосессии без каких-либо меток.

Предоставляйте или не предоставляйте метки, позволяющие iOS давать прерванным приложениям возможность автоматического возобновления воспроизведения аудио, если это необходимо.

Определите, следует ли возобновить аудио, если аудиоперерывание закончилось. Базируйтесь на вашем решении предоставлять пользователю звук в вашем приложении.

- Если ваше приложение отображает элементы управления медиавоспроизведения, которые пользователи используют для воспроизведения или остановки аудио, вам нужно проверить метку `AVAudioSessionInterruptionFlags_ShouldResume` по окончании аудиоперерывания.

Если ваше приложение получает Should Resume flag, вы должны предусмотреть в вашем приложении:

- возобновление воспроизведения звука, если ваше приложение активирует возобновление воспроизведения аудио по завершению прерывания.
- Отсутствие воспроизведения звука, если ваше приложение не активирует воспроизведение аудио по завершении прерывания.

- Если ваше приложение не отображает контроллеры воспроизведения медиа, которые пользователи могут использовать для воспроизведения или остановки аудио, вы должны предусмотреть в вашем приложении возможность всегда возобновлять воспроизведение аудио по окончании прерывания. Вам не требуется осуществлять проверку на наличие меток.

Например, игра, воспроизводящая саундтрек, должна возобновлять воспроизведение после приостановки.

Гарнитурное Медиауправление в Случае Необходимости

Начиная с iOS 4.0, приложения могут получать удаленный контроль над событиями, когда пользователи используют iOS-контроллеры мультимедиа или аксессуар управления (например, контроллеры гарнитуры). Это позволяет вашему приложению принимать

пользовательский ввод, минуя UI, вне зависимости от того, на переднем плане происходит воспроизведение или в фоновом режиме.

В iOS 4.3 и выше приложения могут отправлять видео в AirPlay с поддержкой аппаратных средств, таких как Apple TV, и продолжать воспроизведение при переходе в фоновый режим. Такое приложение может также принимать ввод данных пользователем с помощью удаленного управления событиями, так что пользователи могут управлять воспроизведением видео, в то время как приложение находится в фоновом режиме.

Воспроизведение мультимедиа приложений, в частности, должно адекватно реагировать на события удаленного медиауправления, особенно если воспроизведение аудио или видео осуществляется в то время, как приложение находится в фоновом режиме.

Для выполнения обязанностей, связанных с привилегией воспроизведения медиа, в то время как ваше приложение работает в фоновом режиме, обязательно соблюдайте следующие правила:

Ограничьте право вашего приложения на получение удаленного управления событиями, когда это имеет смысл. Например, если ваше приложение позволяет пользователям читать контент, искать информацию и слушать звук, оно должно принимать удаленное управление пока пользователь находится в звуковом контексте. Когда пользователь покидает контекст аудио, вы должны запретить возможность получения событий. Если ваше приложение позволяет пользователям воспроизводить аудио или видео на AirPlay-совместимом устройстве, оно должно принимать удаленный контроль события, пока воспроизводится мультимедиа. Соблюдение этих принципов позволяет пользователям использовать различные медиаприложения (и управлять ими с помощью гарнитуры управления), когда они в режиме nonmedia вашего приложения.

По возможности, используйте предоставляемые системой контроллеры, чтобы обеспечить поддержку AirPlay. При использовании класса `MPMoviePlayerController` для включения AirPlay-воспроизведения, можно воспользоваться стандартным управлением, что позволяет пользователем выбирать AirPlay-совместимое устройство, находящееся в пределах досягаемости. Или вы можете использовать `MPVolumeView` класс для отображения AirPlay с поддержкой аудио и видео устройств, из которых пользователи могут выбирать. Пользователи привыкли к внешнему виду и поведению этих стандартных элементов управления, поэтому они будут знать, как использовать их в вашем приложении.

Не перепрофилируйте события, даже если они не имеют никакого значения в вашем приложении. Пользователи ожидают, что iOS мультимедиа контроллеры и дополнительные элементы управления функциями последовательны во всех приложениях. Вам не нужно обрабатывать события, которые вашему приложению не требуются, но события, которые вы обрабатываете, должны приводить к восприятию, ожидаемому пользователями. Если вы переопределяете смысл события, вы путаете пользователей и рискуете привести их в недоумение, с которым они не справятся, не покидая приложение.

VoiceOver and Accessibility (Управление Голосом и Доступность)

Управление голосом предназначено для повышения доступности для слепых и пользователей с плохим зрением, а так же для пользователей с определенными проблемами обучения.



Чтобы убедиться, что VoiceOver-пользователи могут использовать ваше приложение, вам придется предоставить некоторую описывающую информацию об окнах и элементах управления в пользовательском интерфейсе. Поддержка VoiceOver не требует изменений визуального оформления пользовательского интерфейса в любом случае.

При использовании стандартных элементов UI в неизменном виде у вас мало (если таковые имеются) дополнительных работ. Чем больше индивидуализированных пользовательских интерфейсов, тем более уникальную информацию необходимо предоставить; используйте VoiceOver для точного описания вашего приложения.

Убедитесь, что вы создали доступ к своему приложению для пользователей, использующих VoiceOver. Это также может увеличить базу ваших пользователей, и это может помочь вам в соблюдении основных принципов доступа, созданных различными руководящими органами.

Edit Menu (Меню Редактирования)

Пользователи могут раскрыть меню редактирования для выполнения таких операций, как Вырезать, Вставить и выбрать в текстовом окне, web-окне или окне изображения.



Вы можете настроить некоторые из функций меню, чтобы дать пользователям больше контроля над содержанием в вашем приложении. Например, вы можете:

- Указать, какие из стандартных команд меню соответствуют текущему контексту;
- Определить положение меню до его появления и тем самым предотвратить закрытие важной части UI вашего приложения;
- Определить объект, который выбран по умолчанию, когда пользователь двойным нажатием вызывает меню.

Вы не можете изменять цвет или форму самого меню.

Для получения информации о том, как реализовать эти действия в коде, см. раздел “Copy and Paste Operations” в iOS Application Programming Guide.

Чтобы гарантировать соответствие вашего приложения пользовательскому ожиданию, вы должны:

Отображать команды, которые имеют смысл в данном контексте. Например, если ни один элемент не выделен, то меню не должно содержать команды “Копировать” или “Вырезать”, поскольку эти функции работают с выделенными элементами. Аналогично, если что-то выбрано, меню не должно содержать “Выбрать”. Если вы поддерживаете меню редактирования в пользовательском окне, вы несете ответственность за то, чтобы команды меню отображались в соответствии с текущим контекстом.

Подогнать отображение меню в макете. iOS отображает меню редактирования выше или ниже курсора или выделения в зависимости от свободного места и места указателя меню, чтобы пользователи могли видеть, как команды меню связаны с содержанием. Можно программно определить положение меню до его появления, так что вы можете предотвратить закрытие важной части пользовательского интерфейса, при необходимости.

Осуществить поддержку жестов, которые пользователи могут использовать для вызова меню. Хотя прикосновение и задержка являются основными жестами пользователей, которые используются для раскрытия меню редактирования, они могут также дважды нажать на слово в текстовом виде, чтобы выбрать слово и открыть меню в то же время. Если вы поддерживаете меню в индивидуализированном окне, реагирование на жесты обязательно. Кроме того, вы можете определить объект, выбираемый по умолчанию, при двойном нажатии пользователем.

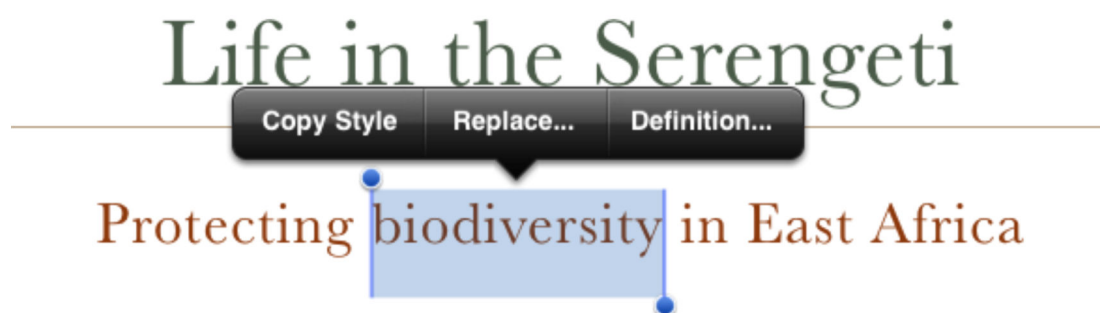
Избегать создания кнопок в интерфейсе, которые выполняют команды, доступные в меню редактирования. Например, лучше позволить пользователям копирование через меню редактирования, нежели предоставить кнопку “Копировать”, иначе пользователи удивятся тому, что одно действие можно выполнять двумя способами в вашем приложении.

Продумать, возможен ли выбор статического текста, если это полезно для пользователя. Например, пользователь может скопировать заголовок изображения, но вряд ли захочет копировать метку элемента вкладки или заголовок экрана, например Accounts. В текстовом окне выбор слова должен быть по умолчанию.

Не создавать кнопки, названия которых можно нажать. Выбор названия кнопки затрудняет для пользователей раскрытие меню редактирования без активации кнопки. В целом, элементы, которые ведут себя как кнопки, не должны быть выделяемыми.

Объединить поддержку отмены и повтора с поддержкой копирования и вставки. Пользователи зачастую хотят иметь возможность отмены последних операций, если они изменили свое решение. Поскольку меню редактирования не требует подтверждения до выполнения своих действий, вы должны дать пользователям возможность отменить или повторить эти действия (чтобы узнать, как это сделать, см. “Undo and Redo” (стр. 95)).

В iOS 4 и выше, вы можете предоставить уникальные, специальные для приложения команды для просмотра в меню редактирования. Следующий пример показывает меню, которое позволяет пользователям копировать стиль в отличие от простого копирования текста.



-----**Примечание:** Если Вам необходимо включить действия, которые используют выделенный текст или объект, способом, выходящим за пределы текущего контекста, то лучше использовать список действий. Например, если вы хотите, чтобы пользователи могли поделиться своим выбором с другими, вы могли бы отобразить **список действий**, в котором перечислены сайты социальных сетей, для выбора и отправки конкретного сайта. Чтобы узнать об использовании руководящих принципов касательно списка действий, см. "Action Sheet" (стр. 128).

Следуйте этим рекомендациям, если вам требуется создать пользовательское меню редактирования элементов.

Создать меню редактирования элементов, редактирующее, изменяющее или иным образом действующее непосредственно по выбору пользователя. Пользователи ожидают стандартное меню редактирования элементов для действий над текстом или объектами в текущем контексте, и лучше всего, когда пользовательские элементы меню ведут себя аналогично.

Отображать список индивидуализированных элементов вместе, после предоставляемых системой. Не перемешивайте ваши индивидуализированные элементы с предоставляемыми системой.

Разумно определить количество уникальных элементов меню. Вы не должны озадачивать своих пользователей слишком большим выбором.

Использовать короткие названия для своих уникальных элементов меню и убедиться в том, что их названия соответствуют их функциям. В общем, названиями элементов должны быть глаголы, описывающие действия, которые должны быть выполнены. Хотя, как правило, следует использовать одно слово в названии элемента с большой буквы, используйте также данный стиль ко всем словам во фразе, если вы используете короткую фразу. (Вкратце, это означает, что вы должны написать каждое слово с большой буквы, за исключением артиклей, соединительных союзов и предлогов, состоящих из четырех и менее букв.)

Undo and Redo (Отменить и Повторить)

Пользователи инициируют операцию отмены путем встряхивания устройства, в котором отображается предупреждение, позволяющую:

- Отменить то, что они только что ввели;
- Повторить отмененный ввод;
- Отменить операцию отмены.

Вы можете поддержать операцию "Отменить" в более общем виде в приложении, указывая при этом:

- Действия, которые пользователи могут отменить или повторить;
- Обстоятельства, при которых ваше приложение должно интерпретировать встряхивание как жест отмены операции;
- Сколько уровней отмены поддерживается.

Чтобы узнать, как реализовать эти функции в коде, см. в разделе "Undo Architecture". Если вы поддерживаете отмену и повтор в своем приложении, выполните следующие рекомендации, чтобы обеспечить хорошее взаимодействие с пользователем.

Предоставьте краткие описывающие фразы, которые будут говорить пользователям именно, что именно они отменяют или повторяют. iOS автоматически предоставляет строки "Undo" и "Redo" (включая пробел после слова) для заголовка кнопки предупреждения об отмене, но вы должны предоставить пару слов, описывающих действия, которые пользователи могут отменить или повторить. Например, вы можете предоставить текст "Удаление Заголовка" или "Изменение Адреса", чтобы создать заголовков кнопки, такой как "Отменить удаление заголовка" или "Повторить изменение адреса". (Обратите внимание, что кнопка "Отмена" в предупреждении не может быть изменена или удалена.)

Избегайте размещения слишком длинного текста. Слишком длинное название кнопки обрезается, что усложняет понимание пользователей. И, так как, что этот текст является заголовком кнопки, используйте стиль заголовка с большой буквы и не используйте пунктуацию.

Избегайте перегрузки жеста встряски. Даже если вы можете программно задать, как ваше приложение будет интерпретировать события встряхивания, вы рискуете ввести в заблуждение пользователей, если они используют данный жест для выполнения различных действий. Анализируйте пользовательское взаимодействие с вашим приложением и избегайте создания ситуаций, в которых пользователи не могут достоверно предсказать результат жеста встряски.

Используйте систему сопровождения кнопок Отмены и Повтора только в случае отмены или повтора фундаментальных задач в вашем приложении. Помните, что жест встряхивания является основным способом для пользователей инициировать отмену и повтор, и что это может ввести в заблуждение, если предлагается два различных способа выполнения одной задачи. Если вы решите, что важно обеспечить явными, специальными элементами отмену и повтор, вы можете разместить предоставляемые системой кнопки в панели навигации. (Чтобы узнать больше об этих кнопках, см. раздел "Standard Buttons for Use in Toolbars and Navigation Bars" (стр. 157).)

Тщательно соотносите возможности отмены и повтора действий непосредственно в пользовательском контексте, а не вне его. Рассмотрим контекст действий, которые могут быть отменены или повторены. В целом пользователи ожидают, что их изменения и действия вступают в силу немедленно.

Keyboards and Input Views (Клавиатуры и Окна Ввода)

Индивидуализированное окно ввода может заменить предоставляемую системой экранную клавиатуру в приложениях, работающих в iOS 3.2 и более поздних. Например, Numbers на iPad предоставляет окно ввода, которое было разработано, чтобы сделать ввод даты и времени простым и эффективным.



Если вы предоставляете индивидуализированное окно ввода, убедитесь, что его функции очевидны для пользователей. Кроме того, убедитесь, что сделанные вами контроллеры в окне ввода выглядят так, что их можно нажимать.

Вы так же можете предоставить дополнительное окно ввода, которое является отдельным окном, располагающимся над клавиатурой (или ваше уникальное окно ввода). Например, в некоторых случаях, Numbers отображаются в дополнительном окне ввода, позволяющем выполнять стандартные или уникальные вычисления по таблице значений.



В iOS 4.2 и более поздних версиях можно использовать стандартную клавиатуру со звуком щелчка для обеспечения звуковой обратной связи, когда пользователи нажимают на уникальные элементы вашего окна ввода. Для того, чтобы узнать, как включить этот звук в ваш код, см. документации по "playInputClick" в описании класса UIDevice.

Примечание: стандартный звук нажатия доступен только для индивидуализированных окон ввода, отображаемых в настоящее время на экране. Пользователи могут отключать все клавиатурные звуки (в том числе те, которые приходят с вашего индивидуализированного окна ввода) в меню "Настройки > Звуки".

Location Services (Службы Определения Местоположения)

Службы определения местоположения позволяют приложениям определять приблизительное географическое местоположение пользователей, направление устройства, которое они указывали, и в каком направлении они движутся. Пользователи ценят возможность автоматически помечать контент своим физическим местоположением или находить друзей, находящихся в настоящее время рядом с ними, но они также ценят возможность отключать такие функции, когда не хотят рассказывать остальным о своем местоположении. (Для того, чтобы узнать, как вставить это в ваше приложение, см. "Location Awareness Programming Guide".)

Когда пользователи отключают Службы определения местоположения и дальнейшее использование функций приложения, требующих их расположения, они видят преду-

предупреждение, которое сообщает о том, что они должны изменить свое предпочтение, прежде чем смогут использовать эту функцию. Предупреждение не позволяет пользователям внести эти изменения в пределах приложения, вместо этого пользователям следует воспользоваться настройками и изменить свои предпочтения. Это гарантирует, что пользователи в полной мере осознают, что они предоставляют общесистемное разрешение использовать информацию об их местоположении.



Следуйте этим советам, чтобы обеспечить лучшее пользовательское впечатление о функции определения местоположения в вашем приложении:

Убедитесь, что пользователи понимают, почему их просят включить службы определения местоположения. Это естественно для пользователей относиться с подозрением к просьбе о доступе к своей личной информации, если они не видят очевидной потребности в этом. Чтобы избежать неудобства пользователей, убедитесь, что предупреждение появляется только при попытке использования функции, которая четко должна знать их расположение. Например, пользователи могут использовать карты при отключенных Службах определения местоположения, но они видят предупреждение, когда они получают доступ к функциям, которые находят и отслеживают их текущее местоположение.

Проверяйте пользовательское предпочтение местоположения во избежание запуска излишних предупреждений. Вы можете использовать программные интерфейсы Core Location, чтобы получить эту настройку (чтобы узнать, как это сделать, см. "Core Location Framework Reference"). Обладая этими знаниями, вы можете вызывать предупреждение как можно ближе к началу работы функции, требующей информацию о местоположении, или, возможно, избежать оповещения в целом.

Показывайте уведомление при запуске вашего приложения, только если ваше приложение не может выполнять основную функцию, не зная местоположения пользователя. Пользователи не будут беспокоиться, потому что они будут понимать, что основная функция вашего приложения зависит от сведений об их расположении.

Избегайте каких-либо программных вызовов, которые инициируют предупреждения, до того, как пользователь на самом деле выбирает функцию, которой необходима эта информация. Таким образом, вы не заставляете пользователей задаваться вопросом, почему приложение требует информацию об их местонахождении, когда они делают что-то, не требующее этого. (Обратите внимание, что получение пользовательского предпочтения не вызывает предупреждений.)

iOS UI Element Usage Guidelines

Руководство по использованию iOS UI-Элементов

В iOS UIKit framework предоставляет широкий ассортимент UI-элементов, которые вы можете использовать в вашем приложении. В процессе дизайна пользовательского интерфейса (UI) вашего приложения всегда помните, что пользователи ожидают уже знакомое им поведение контроллеров и общий вид страниц приложения, основываясь на встроенных приложениях. Если вы используете эти элементы в вашем приложении должным образом, это является существенным преимуществом.

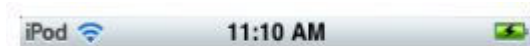
Еще одним плюсом использования стандартных элементов пользовательского интерфейса является то, что они автоматически получают обновления, если iOS предоставляет изменение внешнего вида (полностью уникальные элементы не получают обновлений). Если вы создаете уникальные элементы интерфейса, во всем повторяющие стандартные, кроме внешности, вам следует рассмотреть возможность использования индивидуализации интерфейса стандартных элементов, доступной в iOS5 и выше. При использовании этих API вы можете настроить внешний вид большинства элементов пользовательского интерфейса и по-прежнему получать автоматические обновления их изменений.

Bars (Панели)

Status bar (Статус-панель), Navigation bar (Панель Навигации), Tab bar (Панель Ярлыков) и Toolbar (Панель Инструментов) являются элементами UI, которые имеют определенный вид и поведение в iOS-приложениях. Эти панели не обязательно должны присутствовать в каждом приложении (приложения, которые предоставляют увлекательный опыт (игры) часто не отображают какой-либо из них), но если они все-таки присутствуют, очень важно использовать их должным образом. Панели предоставляют привычные точки опоры для пользователей устройств на iOS-основе, привыкших получать информацию, которую панели отображают и типы функционала, которые они предоставляют.

Status Bar

Status bar (Статус-панель) отображает важную информацию об устройстве и текущей окружающей обстановке.



Константа `UIStatusBarStyle`, описанная в `UIApplication Class`, контролирует стиль Status bar (Статус панели). Чтобы более точно определить стиль Status bar, установите значения в вашем Info.plist файле (Более подробную информацию о содержимом этого файла см. “UIKit Keys”; чтобы узнать, как установить значения, см. Property List Editor Help.)

Внешний вид и Поведение

Status bar (Статус-Панель) всегда располагается в верхнем краю экрана устройства (во всех ориентациях) и содержит информацию, которая необходима пользователю, такую как соединение с сетью, время и дата, заряд батареи.

На iPhone Status bar (Статус-панель) может иметь различные цвета; на iPad Status bar всегда черный.

Руководство

Несмотря на то, что вы не используете Status Bar точно так же, как вы используете другие элементы UI, очень важно понимать его функционирование в вашем приложении.

Подумайте дважды, прежде чем скрывать Status bar, если ваше приложение не является игрой или полноэкранным приложением для просмотра медиа. Несмотря на то, что эти приложения временно скрывают status bar, вы должны понимать последствия этого проектного решения. Временное скрывание status bar означает, что пользователь должен выйти из вашего приложения, чтобы понять, например, нужна ли зарядка для устройства.

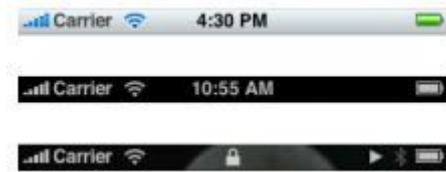
Примечание: большинству iPad-приложений нет необходимости прятать Status bar (Статус-панель), чтобы получить дополнительное пространство, потому что Status bar (Статус-панель) занимает очень не большую часть экрана. На iPad почти незаметный внешний вид Status bar (Статус-панель) не является конкурентом за внимание пользователя с вашим приложением. Маленький размер, вместе с округленными углами в верхней части приложения, создает видимость, что Status bar (Статус-панель) является частью фона устройства.

Рассматривайте необходимость скрывания Status bar (Статус-панель) (и остальных элементов UI приложения) в то время как пользователи находятся в процессе просмотра медиа в полноэкранном режиме. Если вы делаете так, убедитесь, что вы позволяете пользователям восстанавливать Status bar (Статус-панель) и необходимые элементы UI приложения одним нажатием на экран. Несмотря на то, что у вас есть очень важная причина сделать так, лучше избежать назначения индивидуализированного жеста, чтобы снова показать Status bar (Статус-панель), потому что пользователям не очень понравится необходимость узнавать и запоминать данный жест.

Не создавайте уникальные статус панели. Пользователи зависят от согласованности предоставляемой системой статус-панели. Хотя вы можете скрыть статус-панель в вашем приложении, это не подходит для создания уникального пользовательского интерфейса, который занимает это место.

При необходимости, показывайте индикатор активности сети. Индикатор активности сети может появляться на статус-панели, чтобы показывать пользователям уровень доступа к сети. Чтобы изучить применение данного индикатора в вашем коде, см. “Network Activity Indicator” (стр. 139).

На iPhone вы можете определять цвет статус-панели. Вы можете использовать серый цвет (цвет по умолчанию), матовый черный или черный полупрозрачный (который является черным с альфа значением 0.5).



Убедитесь, что выбранный внешний вид статус-панели соответствует остальному внешнему виду вашего iPhone-приложения. Например, избегайте использования полупрозрачной статус-панели, если панель навигации является матово-черной.

На iPhone, установка измененного цвета статус-панели должна быть анимирована.

Примечание: анимация выглядит так, словно старая панель сдвигается, пока не исчезнет с экрана, в то время как новая становится на свое место.

Navigation Bar (Панель Навигации)

Панель навигации дает возможность совершать навигацию через информационную иерархию и, опционально, осуществлять управление содержанием экрана.



Панель навигации содержится в навигационном контроллере, который является объектом, управляющим отображением иерархии индивидуализированных окон. Чтобы узнать больше об описании панели навигации в коде, см. "Navigation Controllers" и UINavigationController Class Reference.

Внешний вид и Поведение

Панель навигации располагается у верхнего края экрана приложения, непосредственно под статус-панелью. Обычно на панели навигации отображается заголовок текущего экрана, отцентрированный по его длине. Когда панель навигации проходит сквозь иерархию информации, пользователь нажимает на кнопку Назад (Back), которая располагается слева от заголовка, чтобы вернуться на предыдущий экран. В другом случае пользователь может нажать на специфические контроллеры (исходя из контента), находящиеся на панели навигации, для управления контентом на экране.

Все контроллеры на панели навигации окружены окантовкой. В iOS это называется окантованный стиль. Если вы располагаете на панели навигации контроллер, у которого отсутствует окантовка, он автоматически конвертируется в стиль с окантовкой.

Панель навигации может быть полупрозрачной или непрозрачной. Если панель прозрачная, верхний край на главном экране соответствует нижнему краю панели

статуса, таким образом, пользователь может видеть содержимое, находящееся за панелью навигации. Если панель непрозрачная, верхний край на главном экране соответствует нижнему краю панели навигации.

На iPhone при изменении ориентации устройства с вертикальной на горизонтальную, может автоматически измениться высота панели навигации. На iPad высота и полупрозрачность панели навигации не изменяется при развороте.

На iPhone панель навигации всегда отображается по всей ширине экрана. На iPad панель навигации может отображаться внутри окна, например, на панели в Split View (Разделенном Окне), которая не растягивается на длину всего экрана.

Руководство

Вы можете использовать панель навигации, чтобы иметь возможность навигации по различным экранам, или предоставить контроллеры, которые управляют элементами на окне.

Используйте заголовок текущего окна как заголовок на панели навигации. При переходе на новый уровень иерархии навигации должны происходить две вещи:

- Заголовок панели должен измениться на заголовок нового уровня.
- Слева от заголовка должна появиться кнопка, которая ведет назад, и она должна быть названа в соответствии с заголовком предыдущего уровня.

Убедитесь, что на панели навигации можно легко прочитать текст. Системный шрифт максимально удобен для прочтения, но вы также можете определить и другой шрифт, если это необходимо.

Используйте панель инструментов вместо панели навигации, если вам необходимо предоставить расширенный набор контроллеров, или если у вас нет необходимости в доступной навигации.

Обдумайте расположение сегментного контроллера на панели навигации на самом верхнем уровне приложения. Это особенно удобно, так как помогает сгладить вашу информационную иерархию и делает поиск более простым для пользователя. Если вы используете сегментный контроллер на панели навигации, убедитесь, что вы выбрали наиболее точные заголовки для названий кнопок, ведущих назад. (Для использования руководства, смотрите “Segmented Control” (стр. 145))

Избегайте загромождения панели навигации дополнительными контроллерами, даже если для них есть достаточно места. Панель навигации должна содержать только заголовок текущего экрана, кнопку Назад, и один контроллер, который управляет контентом окна. Если вместо этого на панели вы используете сегментный контроллер, на нем не должен отображаться заголовок, и он не должен содержать других дополнительных контроллеров.

Используйте кнопки, предоставляемые системой, согласно их задокументированному значению. Для изучения более подробной информации, смотрите “Standard Buttons for Use in Toolbars and Navigation Bars” (стр. 150) Если вы решили создать свои собственные контроллеры для панели навигации, смотрите “Icons for Navigation Bars, Toolbars, and Tab Bars” (стр. 170), чтобы увидеть советы по их дизайну.

При необходимости индивидуализируйте внешний вид панели навигации. Если хотите, чтобы панель навигации выглядела соответственно внешнему виду вашего приложения, вы можете индивидуализировать ее. Например, вы можете предоставить уникальное изображение фона или оттенок панели, также вы можете указать прозрачность. В некоторых случаях хорошей идеей, может быть масштабируемый

размер фонового изображения. Чтобы узнать больше о создании изображений масштабируемого размера, см. раздел “Tips for Creating Resizable Images” (стр. 160).

Убедитесь, что внешний вид уникальной панели навигации соответствует внешнему виду вашего приложения. Если вы используете полупрозрачные панели навигации, не сочетайте их с непрозрачной панелью инструментов. Кроме того, лучше не менять изображение, цвет и прозрачность навигационной панели для различных экранов в одной ориентации.

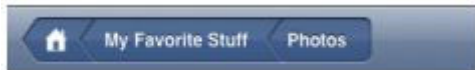
Примечание: Если вы хотите создать панель навигации, которая немного перекрывает содержание по нижнему краю, вы можете предоставить уникальное фоновое изображение, которое должно быть больше, чем стандартная высота панели. В iPhone-приложении вы можете предоставить различные фоновые изображения для разной высоты панели (в iPad –приложении, то же уникальное изображение используется в обеих ориентациях.)

Если вы предоставляете изображение фона для заполнения (taller), лучше создать полупрозрачные изображения (и указать, что панель навигации сама является прозрачной), чтобы пользователи могли видеть содержание за ней.

При необходимости индивидуализируйте внешний вид контроллеров панели навигации. В частности, если вы используете уникальный вид фона для панели навигации, вы должны рассмотреть вопрос о соответствии внешнего вида и панели управления.

Убедитесь, что индивидуализированная кнопка Назад по-прежнему выглядит как кнопка Назад. Пользователи знают, что стандартная кнопка Назад, позволяет им повторить свои шаги по информационной иерархии. Если, например, вы создаете уникальные кнопки Назад, которые не имеют заостренной стороны, пользователи не сразу поймут, что они делают.

Не создавайте мультисегментную кнопку "назад".



Создание мультисегментной кнопки Назад (как на примере выше) вызывает несколько проблем:

- Увеличенная ширина мультисегментной кнопки "назад" не покидает пространство заголовка на текущем экране.
- Нет возможности запрограммировать такую мультисегментную кнопку, чтобы производилась индикация выбранного состояния отдельного сегмента.
- Чем больше сегментов присутствует, тем меньше область, на которую можно нажать, для каждого из сегментов, что вызывает трудности у пользователя при нажатии на какой-то определенный сегмент.
- Проблематично выбрать, какой из уровней отображать в тот момент, когда пользователи проходят глубже по иерархии.

Если вы думаете, что пользователи могут запутаться без мультисегментной кнопки Назад, которая отображается по типу "хлебных крошек", то, вероятно, это означает, что пользователям необходимо слишком далеко погружаться в иерархию информации, чтобы найти то, что им необходимо. Чтобы этого избежать, вам необходимо выровнять иерархию вашей информации.

На iPhone имейте в виду, что при повороте устройства, происходит автоматическое изменение высоты панели навигации. В особенности убедитесь, что

ваши уникальные иконки для панели навигации хорошо подходят для более тонкой панели, которая возникает при горизонтальной ориентации. Не определяйте высоту панели навигации программным способом.

Toolbar (Панель Инструментов)

Панель инструментов содержит контроллеры, которые предоставляют действия, относящиеся к объектам на экране или в окне.



Панель инструментов, как правило, находится в навигационном контроллере, который является объектом, управляющим отображением иерархии уникальных отображений (views). Чтобы узнать больше об описании панели инструментов в вашем коде, смотрите “Displaying a Navigation Toolbar” in View Controller Programming Guide for iOS and UIToolbar Class Reference.

Внешний вид и Поведение

На iPhone панель инструментов всегда располагается у нижнего края экрана или окна, но на iPad она также может находиться и у верхнего края.

Инструменты на панели инструментов отображаются на одинаковом расстоянии относительно друг друга по всей ширине панели инструментов. Установленный набор инструментов на панели инструментов может изменяться от окна к окну, потому что они всегда специфичны для определенного контекста в текущем окне.

На iPhone изменение ориентации устройства из вертикальной в горизонтальную может автоматически изменить высоту панели инструментов. На iPad высота и полупрозрачность панели инструментов не изменяется при вращении устройства.

Руководство

Используйте панель инструментов, чтобы предоставить набор действий, которые пользователь может использовать в текущем контексте.

Используйте панель инструментов, чтобы предоставить пользователям выбор часто используемых команд, которые имеют смысл в текущем контексте.

Альтернативный вариант – расположить части контроллера на панели инструментов, чтобы дать пользователям доступ к различным перспективам данных вашего приложения или различным режимам приложения (для изучения руководства, смотрите “Segmented Control” (стр. 145)).

При необходимости индивидуализируйте внешний вид панели инструментов. Если вы хотите, привести в соответствие с общим видом вашего приложения панель инструментов, вы можете указать фоновое изображение или оттенок и прозрачность. В некоторых случаях хорошей идеей, может быть масштабируемое изображение фона.

Чтобы узнать больше о масштабируемом размере изображения, см. раздел “Tips for Creating Resizable Images” (стр. 160).

Убедитесь, что ваша индивидуализация панели инструментов соответствует виду вашего приложения. Если вы используете полупрозрачные панели инструментов, не сочетайте их с непрозрачной панелью навигации. Кроме того, лучше избегать изменения внешнего вида панели инструментов для различных экранов в одной ориентации.

Примечание: Если вы хотите создать панель инструментов, которая немного перекрывает основное содержание, вы можете предоставить уникальное фоновое изображение, которое должно быть больше, чем стандартная высота панели. В iPhone-приложении вы можете предоставить различные фоновые изображения для разной высоты панели (в iPad-приложении то же уникальное изображение используется в обеих ориентациях.)

Если вы предоставляете изображение фона для заполнения (taller), лучше создать полупрозрачные изображения (и указать, что панель навигации сама является прозрачной), чтобы пользователи могли видеть содержание за ней.

Поддерживайте область в размере не менее, чем 44x44 px для каждой кнопки на панель инструментов. Если вы расположите кнопки на панели инструментов слишком близко друг к другу, пользователям, скорее всего, будет трудно нажимать именно на ту кнопку, которая им нужна.

Используйте Toolbar-кнопки, предоставленные системой, согласно их задокументированному значению. Смотрите “Standard Buttons for Use in Toolbars and Navigation Bars” (стр. 150) для более подробной информации. Если вы решили создать свои собственные Toolbar-кнопки, смотрите “Icons for Navigation Bars, Toolbars, and Tab Bars” (стр. 170), где вы сможете найти советы по их дизайну.

При необходимости индивидуализируйте внешний вид панели инструментов. Если вы индивидуализируете внешний вид панели инструментов, вы можете рассмотреть вопрос о соответствии внешнего вида и панели инструментов. Вы также можете настроить выбранный вид элементов, чтобы они хорошо смотрелись на фоне индивидуализированной панели инструментов.

Попытайтесь избежать смешения стиля без окантовки и окантованного стиля кнопок панели инструментов на одной и той же панели инструментов. Вы можете использовать любой из этих стилей, но их совмещение обычно не выглядит удачно.

На iPhone имейте в виду, что высота панели инструментов автоматически изменяется при изменении ориентации устройства. В особенности убедитесь, что ваши уникальные Toolbar-иконки подходят к меньшей высоте панели инструментов в горизонтальной ориентации. Не определяйте высоту панели инструментов программным способом; вместо этого вы можете воспользоваться константами UIBarMetrics, чтобы убедиться, что содержание вписывается.

Tab Bar (Панель Ярлыков)

Tab Bar (Панель Ярлыков) дает пользователям возможность переключаться между различными подзадачами, экранами или режимами.



Панель ярлыков находится в контроллере панели ярлыков, который является объектом, управляющим отображением ваших уникальных представлений (views). Чтобы узнать больше об описании панели ярлыков в вашем коде, см. раздел “Tab Bar Controllers” и UITabBar Class Reference.

Внешний вид и Поведение

Панель ярлыков находится у нижнего края экрана и должна быть доступна из любого экрана в приложении. На панели ярлыков отображаются иконки и текст на ярлыках; все они одинаковы по ширине и отображаются на черном фоне. Когда пользователь выбирает ярлык, например “Найти на YouTube”, ярлык отображается на более светлом фоне (который известен, как образ индикатора выбора) и его иконка получает голубое свечение.



На iPhone на панели ярлыков могут отображаться не более чем 5 ярлыков в одно и то же время; если в приложении более 5-и ярлыков, на панели ярлыков отображаются только 4 из них и добавляется ярлык More (Больше), который открывает дополнительные ярлыки в списке. На iPad на панели ярлыков может располагаться более чем 5 ярлыков.

На Tab bar-ярлыке может отображаться бэйдж (красный овал, который содержит белый текст, либо цифры, либо восклицательный знак), который передает информацию, специфичную для данного приложения.

Панель ярлыков не изменяет свою прозрачность или высоту вне зависимости от ориентации устройства.

Руководство

Используйте панель ярлыков, чтобы предоставить пользователям доступ к различным ракурсам в том же самом наборе данных или различным подзадачам, относящимся к общему функционированию вашего приложения. Когда вы используете панель ярлыков, следуйте следующим инструкциям:

Не используйте панель ярлыков, чтобы предоставить пользователям контроллеры, которые взаимодействуют с элементами в текущем режиме или текущем экране. Если вам необходимо предоставить контроллеры для ваших пользователей, используйте вместо этого панель инструментов (для изучения руководства, см. “Toolbar” (стр. 104)).

В общем, используйте панель ярлыков, чтобы организовать информацию на уровне всего приложения. Панель ярлыков хорошо подходит для использования на основном экране приложения, потому что это хороший способ выровнять иерархию вашей информации и предоставить доступ к нескольким категориям информации или режимам в одно и то же время.

Не удаляйте ярлыки, когда их функции недоступны. Если ярлык представляет собой часть вашего приложения, которая недоступна в данном контексте, лучше отобразить этот ярлык выключенным, чем удалить его. При удалении ярлыка в некоторых случаях, но не во всех, вы делаете свой пользовательский интерфейс приложения нестабильным и непредсказуемым. Лучшим решением будет убедиться, что ваши вкладки всегда включены, но объяснить, почему содержимое ярлыка недоступно. Например, если пользователь не имеет никаких песен на iOS-устройстве, Music-приложение не отключает Song-ярлык, вместо Song-ярлыка отображается экран, который объясняет, как загрузить песни.

На iPad вы должны использовать панель ярлыков на панели в Split View (Разделительное окно) или на поповере, если при нажатии на ярлыки происходит переключение или фильтрация контента внутри окна. Однако достаточно часто это выглядит лучше с использованием сегментного контроллера на нижнем краю поповера или панели Split View, потому что внешний вид сегментного контроллера лучше сочетается с поповером или внешним видом Split View. (Для более подробной информации об использовании сегментного контроллера смотрите “Segmented Control” (стр.145).)

Обдумайте использование иконки бэйджа на панель ярлыков для ненавязчивой коммуникации. Вы можете отобразить бэйдж на иконке панели ярлыков, чтобы он служил индикатором появления новой информации, относящейся к данному окну или режиму.

Используйте иконки панели ярлыков, предоставленные системой, согласно их задокументированному значению. Для более подробной информации смотрите “Standard Icons for Use in Tab Bars” (стр. 153) Если вы решили создать ваши собственные Tab bar-иконки, смотрите “Icons for Navigation Bars, Toolbars, and Tab Bars” (стр. 170), для советов по их дизайну.

В случае необходимости индивидуализируйте внешний вид панели ярлыков. Например, можно указать уникальный цвет для панели ярлыков и иконок, как для иконок, предоставляемых системой, так и для уникальных шаблонов изображения. См. раздел “Icons for Navigation Bars, Toolbars, and Tab Bars” (стр. 170). Можно также указать фоновое изображение для панели ярлыков (заметим, что зачастую хорошо предоставить масштабируемый размер изображений, см. раздел “Tips for Creating Resizable Images” (стр. 160)). Наконец, вы можете также предоставлять различные оттенки для выбранных выделений иконки ярлыка (обратите внимание, что это состояние существует отдельно от изображения индикатора выбора). Если вы не предоставите уникальный цвет выбора, iOS применяет стандартное голубое свечение для выбранного ярлыка панели ярлыков, а если вы предоставляете уникальный цвет выбора, iOS применяет только свечение.

Если нужно, предоставляйте индивидуализированный образ индикатора выбора. В частности, если вы предоставите уникальный оттенок на панели ярлыков, вы можете определить выбор изображений индикатора, который соответствует ему. Вы можете предоставить один фиксированный размер для применения к выбранному ярлыку (вы не можете указывать несколько изображений индикатора выбора для разных ярлыков). Кроме того, ваше изображение должно быть полупрозрачным, потому что iOS рисует изображение индикатора выбора за иконкой панели.

На iPad избегайте перенасыщения панели ярлыков слишком большим количеством ярлыков. Расположение слишком большого количества ярлыков на панели ярлыков, может вызвать сложность для пользователя при нажатии именно на тот, который ему нужен. Также с каждым дополнительным ярлыком, который вы отображаете, вы увеличиваете сложность вашего приложения. В общем, пытайтесь ограничить количество ярлыков в главном окне или на правой панели в Split View

(Разделительное окно) до семи. На поповере или на левой панели в Split View (Разделительное окно) нужно ограничить количество до пяти ярлыков.

На iPad избегайте создания ярлыка More (Больше). В iPad-приложении экран, который "прикован" только к списку дополнительных ярлыков - это плохое использование пространства.

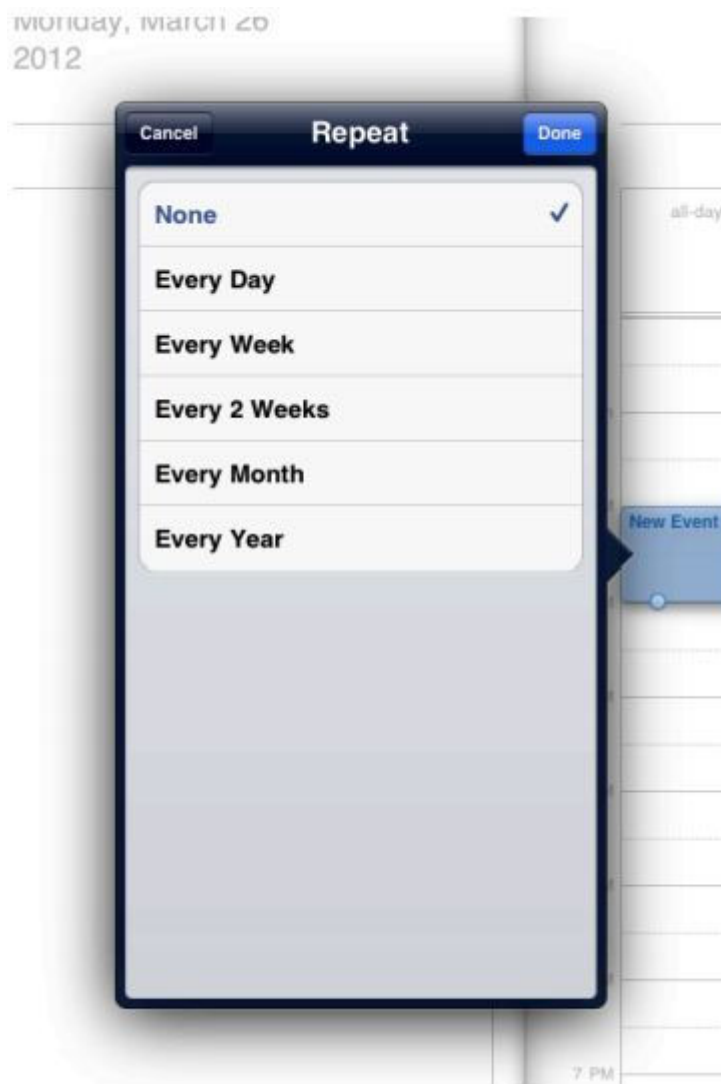
На iPad отображайте те же самые ярлыки в каждой ориентации, чтобы увеличить визуальную стабильность вашего приложения. В вертикальной ориентации рекомендуется использовать семь ярлыков по всей ширине экрана. В горизонтальной ориентации вы должны отцентровать те же самые ярлыки по ширине экрана. Данное руководство также предлагает использование панели ярлыков внутри панели Split View (Разделительное Окно) или поповера. Например, если вы используете панель ярлыков на поповере в вертикальной ориентации, будет хорошо отобразить те же самые ярлыки на левой панели Split View (Разделительное Окно) в горизонтальной ориентации.

Content Views (Контентные Окна)

iOS предоставляет несколько окон, которые предназначены для отображения уникального контента приложения. Каждое окно обладает определенным набором свойств и поведений, что делает их специально предназначенными для отображения определенных типов информации.

Поповер (только для iPad)

Поповер - это временное окно, которое может открываться, когда пользователь нажимает на контроллер или на область внутри экрана.



Чтобы узнать больше об описании попувера в вашем коде, смотри “Popovers” в View Controller Programming Guide для iOS и UIPopoverController Class Reference.

Важно: Попуверы доступны только для iPad-приложений.

Внешний вид и Поведение

Попувер - это отдельное окно, которое перекрывает контент экрана. Он всегда отображает стрелку, которая служит индикатором точки, из которой он появляется. Попувер может содержать разнообразные объекты и окна, такие как:

- Таблица, изображение, карта, текст, web или индивидуализированные окна;
- Панели навигации, панели инструментов или панели ярлыков;
- Контроллеры или объекты, которые взаимодействуют с объектами в текущем окне приложения

В iPad-приложениях, список действий всегда отображается внутри попувера.

Руководство

Вы можете использовать поповер, чтобы:

- Отображать дополнительную информацию или список пунктов, относящихся к фокусному (или выбранному) объекту.
- Отображать контент на левой панели в приложении, которое базируется на Split View (Разделительное Окно) в вертикальной ориентации. Если вы делаете так, убедитесь, что предоставляете необходимый заголовок кнопке, которая отображает поповер, желательно на панели навигации или панели инструментов сверху экрана.
- Отображать список действий, который содержит короткий список опций, непосредственно относящихся к чему-либо на экране.

Избегайте предоставления кнопки "dismiss popover" ("убрать поповер"). Поповер должен автоматически закрыться, когда его присутствие больше не является необходимым. Чтобы определить, что присутствие поповера больше не является необходимым, руководствуйтесь следующими сценариями:

- Когда функция поповера - это предоставление набора опций или предметов, которые влияют на основное окно, он должно закрыться, как только пользователь сделает выбор. Это поведение очень похоже на поведение меню в компьютерном приложении. Примечание: это поведение также применимо к поповеру, который содержит только лист действий: как только пользователь нажимает на кнопку на листе действий, поповер должен закрыться.

Иногда имеет смысл предоставить поповер, который содержит функции, влияющие на основное окно, но не очень близко к тому месту, где пользователи делают свой выбор. Вы, возможно, захотите сделать это при проверке поповера, потому что пользователи могут захотеть сделать дополнительный выбор или изменить атрибуты текущего выбора.

Поповер, который предоставляет меню или проверяет функциональность, должен закрываться в тот момент, когда пользователи нажимают где-либо за пределами его границ, включая контроллер, который открывает поповер. В поповере, который предоставляет меню выборов, это означает, что пользователь решил не делать выбора (и таким образом основное окно остается без изменений). В поповере, который содержит Список Действий, это происходит при нажатии на кнопку Cancel (Отмена).

- Если поповер делает доступной задачу, то было бы хорошо отобразить кнопки, которые завершают или отменяют задачу и одновременно закрывают поповер. В общем, поповеры, которые делают доступной задачу, отображают кнопку Done и кнопку Cancel. Эти кнопки помогают напомнить пользователям, что они находятся в среде редактирования и позволяют им ясно придерживаться или отказаться от ввода своих данных. Когда пользователи нажимают на какую-либо кнопку, поповер должен закрыться.

Если это имеет смысл в вашем приложении, вы можете предупредить пользователей от закрытия такого поповера, когда они нажимают за пределами его границ. Это хорошая идея, если очень важно, чтобы пользователи завершили или ясно отказались от задачи. С другой стороны, вы должны сохранить данные, которые они ввели, когда они нажимают за пределами границы поповера, так же, как если бы они нажали Done.

В общем, сохраняйте работу пользователей, когда они нажимают за пределами границ поповера. Так как для поповера нет четкого осознания для его закрытия,

пользователи могут закрыть его по ошибке. Вы должны сбросить данные об их действиях, только если они нажали на кнопку Cancel (отмена).

Убедитесь, что стрелка поповера указывает наиболее точно на элемент, который его открывает. Это помогает пользователям запомнить, откуда открылся поповер и какая задача либо объект ассоциируется с ним.

Убедитесь, что пользователи могут использовать поповер, не видя содержания контента позади него. Поповер скрывает контент позади него и пользователи не могут перенести поповер в другое место.

Убедитесь, что в одно и тоже время отображается только один поповер. Вы не можете отображать более одного поповера (или уникального окна, которое сделано так, что похоже по дизайну и по поведению на поповер) в одном и том же месте. В особенности вы должны избегать отображения каскада или иерархии поповеров одновременно, где каждый поповер всплывает из предыдущего.

Не отображайте модальное окно (modal view) поверх поповера. Ничего, кроме предупреждения, не может быть отображено поверх поповера.

По возможности, позволяйте пользователям закрыть один поповер и открыть новый одним нажатием на экран. Данное поведение особенно желательно, когда несколько различных кнопок открывают поповер, чтобы предотвратить дополнительные нажатия.

Избегайте создания слишком большого поповера. Поповер не должен перекрывать весь экран. Вместо этого он просто должен быть достаточно большим, чтобы отображать свое содержание и по-прежнему указывать на то место, откуда он появился.

В идеале ширина поповера должна быть минимум 320 пикселей, но не более 600 пикселей. Высота поповера может быть любой, и вы можете использовать это для отображения длинного списка. Хотя вы должны избегать прокрутки в поповере, который предоставляет задачу или Список Действий.

Примечание: Система может отрегулировать высоту и ширину поповера, чтобы убедиться, что он хорошо смотрится на экране.

В общем, используйте стандартные UI-контроллеры и экраны внутри поповера. В общем, поповеры выглядят наилучшим образом, и наиболее просты для понимания пользователей, когда они содержат стандартные контроллеры и экраны.

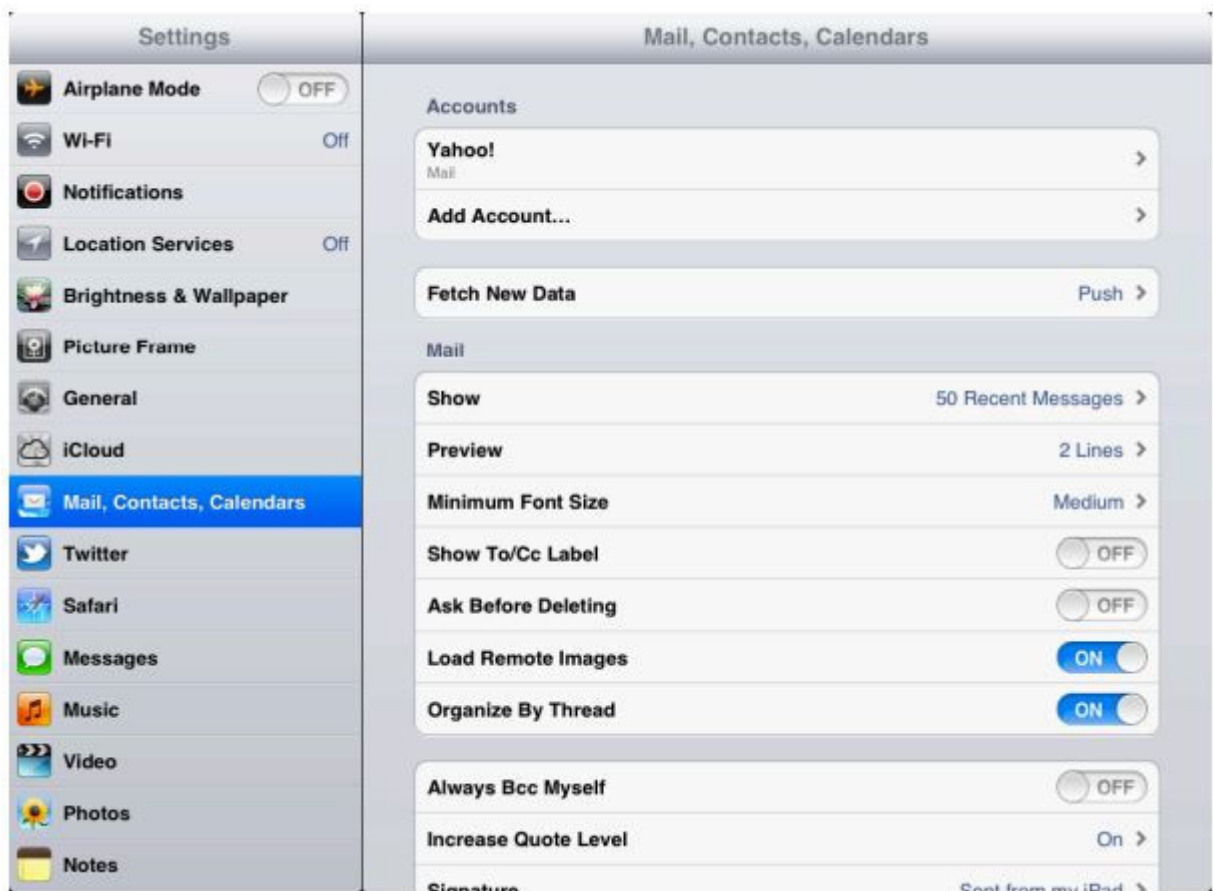
При необходимости индивидуализируйте внешний вид поповера. Хотя достаточно легко настроить визуальные аспекты поповера с помощью `UIPopoverBackgroundView` API, важно избегать создания дизайна, который люди не могли воспринимать как поповер. Если вы измените внешний вид поповера очень сильно, пользователи не смогут рассчитывать на их предыдущий опыт, который они могли бы использовать для понимания вашего приложения. (Если вы хотите предоставить масштабируемый размер изображения фона, см. раздел *Tips for Creating Resizable Images* (стр. 160) для получения дополнительной информации о том, как это сделать.)

Будьте осторожны при комбинировании уникального цвета фона или текстуры со стандартными контроллерами и экранами. Убедитесь, что стандартные элементы UI выглядят хорошо и просты в прочтении, когда они располагаются поверх вашего уникального фона.

Если нужно, измените размер поповера в тот момент, когда его видно. Возможно вы захотите изменить размер поповера если вы используете его, чтобы отображать одну и ту же информацию как в кратком виде, так и в развернутом. Если вы изменяете размер поповера в тот момент, когда он виден, вы можете добавить анимацию данного изменения. Анимация изменения не заставляет пользователей думать, будто новый поповер должен заменить предыдущий.

Split View (Разделительное Окно) (Только для iPad)

Split View - это полноэкранное окно, которое состоит из двух размещенных рядом панелей



Разделительное окно содержится в контроллере разделительного окна, который является объектом, управляющим отображением панелей разделительных окон. Чтобы узнать больше об описании Split View в вашем коде, смотрите [UISplitViewController Class Reference](#).

Важно: Split View доступен только для iPad-приложений.

Внешний вид и Поведение

Ширина левой панели в Split View фиксирована и составляет 320 px во всех ориентациях. Пользователи не могут изменять размер обеих панелей в Split View.

Примечание: Хотя левая панель всегда называется мастер-панелью, а правая панель часто называется детальной панелью, эти отношения не влияют на код.

Обе панели могут содержать широкий набор объектов и полей, такие как:

- Таблицы, изображения, карты, текст, web или уникальные поля;
- Панель навигации, панель инструментов или панель ярлыков.

Руководство

Вы можете использовать Split View чтобы отображать стабильную информацию на левой панели и изменяемые дела или зависимую информацию на правой панели. В этом шаблоне дизайна, когда пользователи выбирают информацию на левой панели, правая панель должна отображать информацию, которая с ней соотносится. (Вы ответственны за то, чтобы это было реализовано в коде.)

Иногда, когда приложение использует Split View в горизонтальной ориентации, оно отображает контент левой панели на поповер при развертке в вертикальную ориентацию. Однако вы не обязаны следовать данному шаблону. Если это имеет смысл в вашем приложении, вы можете сделать дизайн вашего UI, так чтобы отображать совмещенные панели во всех ориентациях.

Избегайте создания правой панели меньшей ширины, чем левой панели. Хотя ширина правой панели определяется на ваше усмотрение, не устанавливайте ее ширину менее 320 пикселей (ширина левой панели) - это выглядит не удачно.

Избегайте отображения панели навигации на обеих панелях одновременно. Если вы сделаете так, определение отношений между двумя панелями будет дополнительной сложностью для пользователей.

В общем, оставляйте неизменным текущий выбор на левой панели. Это поведение помогает пользователям понять взаимоотношение между предметом на левой панели и содержанием на правой панели. Это является очень важным, потому что контент на правой панели может быть изменен, но он всегда должен соотноситься с предметами, выбранными на левой панели.

Table View (Окно Таблицы)

Окно Таблицы представляет собой данные, составленные в список с одной колонкой и множеством строк.

F	Q
Sarah Fahrzad	A
	B
	C
Rachael Falworth	D
	E
	F
Jose Fargo	G
	H
	I
Jim Ferris	J
	K
	L
	M
Kimmie Fong	N
	O
	P
Lynn Foote	Q
	R
	S
G	T
Pete Gardner	U
	V
	W
	X
Monique Gaspard	Y
	Z
	#

Чтобы узнать больше об описании Окна Таблицы в вашем коде, смотрите Table View Programming Guide for iOS and UITableView Class Reference.

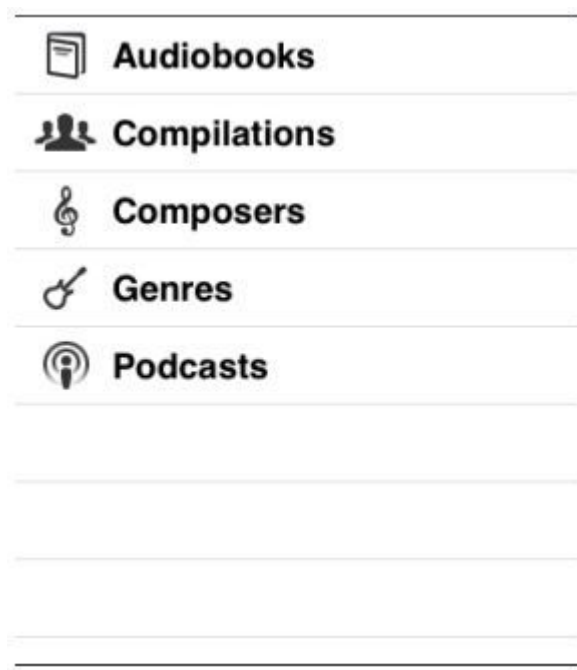
Внешний вид и Поведение

Окно таблицы отображает данные в строках, которые могут быть разделены на секции или выделены в группы. Пользователь может прокручивать вниз и вверх, переходя по строкам или группам строк. Пользователь нажимает на строку таблицы, чтобы выбрать ее и использовать контроллеры Окна Таблицы, чтобы добавить или удалить строки, выбрать несколько строк, просмотреть более подробную информацию о предмете на строке или открыть другое Окно Таблицы. Когда пользователь нажимает на выбираемый предмет - строка таблицы быстро подсвечивается.

Если выбор строки приводит к переходу на новый экран, то выбранная строка быстро подсвечивается и появляется новый экран. Когда пользователь переходит к предыдущему экрану, изначально выбранная строка снова быстро подсвечивается, чтобы напомнить пользователю его предыдущий выбор (он не остается подсвеченным).

iOS определяет два стиля окна таблицы, которые отличаются, в основном только внешним видом.

Простая таблица отображает строки, которые проходят от одного края экрана до другого края (по всей длине экрана). Строки могут быть разделены секциями с названиями, и с правого края, на всю его длину, может стоять рубрикатор. Header может появиться перед первым пунктом в разделе, а footer после последнего раздела в списке. По умолчанию фон строки белый.





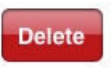
Сгруппированные таблицы отображают группы строк, которые находятся во вкладке от края экрана. Сгруппированное окно таблицы содержит как минимум одну группу пунктов в списке (один список пунктов на строке) и каждая группа всегда содержит как минимум один пункт. Группа может предваряться текстом на header и внизу может быть текст footer. По умолчанию группы отображаются на голубом в полосу фоне; внутри группы фон строки белый. Сгруппированное Окно Таблицы не включает в себя алфавитный указатель.



iOS включает в себя некоторые **элементы окна таблицы**, которые могут увеличивать функциональность окон таблицы. Если не написано обратное, данные элементы можно использовать только с окнами таблицы.

Таблица 7-1 Элементы окна таблицы

Элемент	Название	Описание
	Checkmark (Галочка)	Говорит о том, что строка выбрана
	Disclosure indicator (Индикатор Раскрытия)	Отображает другую таблицу, ассоциирующуюся со строкой
	Detail disclosure button (Кнопка Раскрытия Деталей)	Отображает дополнительные детали о строке в новом окне (для дополнительной информации как использовать этот элемент за пределами таблицы, смотрите “Detail Disclosure Button” (стр 126))
	Row Reorder (Перемещение Порядка Строк)	Показывает, что строка может быть перетянута в другое месторасположение в таблице

Элемент	Название	Описание
	Row Insert (Вставить строку)	Добавляет новую строку в таблицу
	Delete Button Control (Контроллер кнопки удаления)	В редактируемом контексте открывает и скрывает кнопку удаления строки
	Delete Button (Кнопка Удаления)	Удаляет строку

iOS 3 и более поздние версии определяет четыре стиля ячеек, которые принадлежат к большинству шаблонов строк таблицы в простых и сгруппированных таблицах. Каждый стиль ячейки наилучшим образом подходит для отображения различного типа информации.

Примечание: Эти стили применяются программно к ячейке окна таблицы, которая является объектом, сообщаящим таблице, как отображать ее строки.

По умолчанию (UITableViewCellStyleDefault). По умолчанию стиль ячейки включает в себя опциональное изображение с левого края строки, следующее после черного заголовка, выровненного по левой стороне.



В стиле ячейки по умолчанию прилагается текст, который представляет собой название пункта или заголовок. Этот текст выровнен по левой стороне, и его очень легко просмотреть. Это делает стиль ячейки по умолчанию очень хорошим для отображения списка пунктов, которые не должны быть перегружены дополнительной информацией.

Подзаголовок (UITableViewCellStyleSubtitle). Стиль подзаголовка включает в себя опциональное изображение с левого края строки, следующее за выровненным по левой стороне текстом на одной линии и выровненным по левой стороне подзаголовком на линии ниже. Заголовок написан черным, а подзаголовок меньшим шрифтом и серым цветом.



В стиле подзаголовка ячейки выделение текста применяется, чтобы предоставить название пункта или заголовок, несмотря на то, что внешний вид подзаголовка (дополнительный текст) содержит дополнительную информацию относительно пункта списка. Выравнивание текстов по левому краю делает список очень простым для просмотра. Это стиль ячейки хорошо подходит, когда список пунктов выглядит одинаково, потому что пользователи используют дополнительную информацию в дополнительном тексте, чтобы различить одинаково названные пункты.

Значение 1 (UITableViewCellStyleValue1). Стиль со значением 1 отображает выровненный по левому краю заголовок в черном цвете на одной и той же линии и выровненный по правому краю подзаголовок в меньшем размере шрифта и голубым цветом. Изображения не очень хорошо подходят к этому стилю.



В стиле ячейки со значением 1 внешний вид текста предоставляет имя пункта или заголовок, несмотря на то, что внешний вид подзаголовка (дополнительный текст) содержит дополнительную информацию относительно пункта списка.

Выравнивание по левому краю и шрифт текста помогает пользователям просматривать список, чтобы найти необходимый пункт, а выравнивание по правому краю дополнительного текста притягивает их внимание к дополнительной информации, которую он предоставляет. Этот стиль ячейки хорошо подходит для отображения текущего значения пункта, возможно выбранного из подсписка.

Значение 2 (UITableViewCellStyleValue2). Стиль со значением 2 отображает выровненный заголовок, набранный мелким шрифтом голубого цвета, следующий за находящимся на той же линии выровненным по левому краю подзаголовком, набранным крупным шрифтом черного цвета. Изображения не подходят к этому стилю.



В стиле ячейки со значением 2 выровненная по правому краю сдержанная ширина и шрифт текста подразумевают, что более важная информация находится в дополнительном тексте, выровненном по левому краю.

В этом шаблоне, тексты выровнены по отношению к друг к другу, в том же самом месте расположения в каждой строке. Это создает четкое, вертикальное поле между текстом заголовка и дополнительным текстом в списке, что помогает пользователям сфокусироваться на первых словах дополнительного текста.

Примечание: Все четыре стандартных стиля ячейки также позволяют добавлять такие элементы окна таблицы, как галочка или индикатор раскрытия. Добавление этих

элементов уменьшает ширину ячейки, доступную для заголовка и подзаголовка.

Руководство

Вы можете использовать Окно Таблицы, чтобы четко и эффективно отображать большие или малые объемы информации. Например, вы можете использовать Окно Таблицы, чтобы:

Предоставлять список опций, из которых пользователь может выбрать. Вы можете использовать галочку (значок выбора), чтобы показать пользователям опцию (или опции), которые выбраны в списке на данный момент.

Чтобы отобразить список выборов, которые видят пользователи, когда они нажимают на пункт в окне таблицы, вы можете использовать либо простое, либо сгруппированное окно таблицы. Чтобы отобразить список выборов, который видят пользователи, когда они нажимают на кнопку или другой элемент UI, который не находится в строке таблицы, используйте простой стиль.

Отображать иерархическую информацию. Простой стиль таблицы очень хорошо подходит для отображения информационной иерархии. Каждый пункт списка может вести к различным множествам информации, отображенным в другом списке. Пользователи следуют через иерархию, путем выбора пункта в каждом последующем списке. Индикатор раскрытия сообщает пользователям, что нажатие где-либо в строке открывает множество новой информации в новом списке.

Отображать схематически сгруппированную информацию. Оба стиля окна таблицы позволяют вам предоставлять контент путем предоставления текста на header и footer, между разделами информации. Закругленные углы сгруппированного стиля разделяют группы информации так, что это легко различимо, даже когда пользователи просматривают быстро.

Отображать информацию, которая проиндексирована, чтобы способствовать поиску. Простой стиль поддерживает окно оглавления, что помогает пользователям быстро найти то, что им нужно. Оглавление состоит из колонки (обычно с буквами алфавита), которая прикреплена с правой стороны экрана. Пользователи нажимают или прокручивают к данным оглавления, чтобы открыть соответствующую область списка.

Если вы используете оглавление, избегайте использования элементов окна таблицы, которые отображаются с правой стороны таблицы (таких как индикатор раскрытия), потому что эти элементы пересекаются с оглавлением.

Всегда предоставляйте обратную связь, когда пользователи выбирают пункт списка. Пользователи ожидают, что строка таблицы быстро подсветится, когда они нажимают на выбранный пункт. После нажатия пользователь ожидает, что сразу появится действие: или появится новое окно, или на строке отобразится галочка, которая будет показывать, что этот пункт списка выбран или доступен.

В редких случаях строка может оставаться подсвеченной, когда на том же самом экране отображаются дополнительные детали или контроллеры, которые относятся к этому пункту. Однако, это не одобряется, потому что очень трудно одновременно отображать список выборов, выбранный пункт и детали или контроллеры без создания перегруженной схемы экрана.

Следуйте следующим инструкциям при использовании окна таблицы:

Анимлируйте изменения, которые производит пользователь со списком пунктов, чтобы предоставить обратную связь и ощущение прямого взаимодействия пользователя

с приложением. В Настройках (Settings), например, когда пользователи отключают настройки автоматической даты и времени (при выборе OFF в General > Date & Time > Set Automatically), плавно выпадает список групп, чтобы отобразить два новых пункта, Time Zone и Set Date & Time.

Если содержание таблицы очень большое или сложное, избегайте ожидания того, чтобы все данные были доступны перед тем, как их отображать. Вместо этого сразу заполните строки, отображающиеся на экране, текстовыми данными и отображайте более сложные данные (такие как изображения) как только они станут доступными. Эта техника сразу дает пользователям полезную информацию и увеличивает реакцию вашего приложения.

Отображайте "старые" данные в то время, пока ждете загрузки новых обновленных данных. Если ваше приложение отображает данные, которые нечасто изменяются, вы, возможно, примените эту технику - и, таким образом, пользователи могут сразу увидеть что-то полезное. Эта техника не рекомендуется для приложений, которые содержат часто обновляющиеся данные. Прежде чем вы решите сделать это, определите, как часто изменяются данные и насколько сильно пользователи зависят от быстрого просмотра новой информации.

Если данные сложные или медленно подгружаются, предоставьте пользователям какой-то знак, что загрузка данных находится в процессе. Если таблица содержит только сложные данные, то будет очень сложно быстро отобразить что-то полезное. В этом редком случае очень важно избегать отображения пустых строк, потому что это может вызвать предположение, что приложение "зависло". Вместо этого таблица должна отображать крутящийся индикатор активности, с каким-либо информативным текстом, например "Loading", выровненным по центру экрана. Это предоставляет пользователям обратную связь, дающую им знать, что процесс продолжается.

Избегайте различной высоты строк в простой таблице. Различная высота строк доступна в сгруппированной таблице, но в простой таблице это может привести к беспорядку.

Не используйте белые участки изображения, которые должны выглядеть как прозрачные. Фон строки таблицы не чисто белый, поэтому белые участки на изображении будут белыми; они не будут сливаться с фоном и прозрачностью. Если вы хотите, чтобы некоторые области вашего изображения выглядели прозрачными, не забудьте сделать их таковыми.

В случае необходимости используйте уникальное название кнопки 'Delete'. Если это поможет пользователям лучше понять, как ваше приложение работает, вы можете создать заголовок, чтобы заменить предоставляемый системой заголовок 'Delete'.

В общем, используйте сгруппированную таблицу для стилей ячеек со значением 1 и 2. Хотя вы можете использовать любой стиль ячейки с любым типом таблицы, просто стиль ячеек со значением 1 и 2 - наиболее подходит для сгруппированных таблиц. Например, "Settings" используют стиль ячейки со значением 1.



iPhone Contacts используют стиль ячейки со значением 2:



Как можно тщательнее проверьте и убедитесь, что ваши тексты являются лаконичными, чтобы избежать сокращений. Сокращенные слова и фразы могут вызывать затруднения в прочтении и понимании. Сокращение текста происходит автоматически во всех стилях ячеек таблицы, но это может вызвать некоторые

проблемы, в зависимости, от того какой стиль ячейки вы используете и в каком месте текста происходит сокращение.

- В стиле ячейки по умолчанию лучше всего использовать короткие тексты. Если невозможно избежать сокращения, постарайтесь убедиться, что наиболее важная информация хранится в первых нескольких словах.
- В стиле ячейки подзаголовка также лучше всего использовать короткие тексты. Если невозможно избежать сокращения, сфокусируйтесь на вставке наиболее важной информации в первые несколько строк. Если дополнительный текст сокращен, пользователи вряд ли особенно обратят на это внимание, потому что они просматривают этот текст как дополнительный.
- В стиле ячейки со значением 1 сокращения текста, возможно, затруднительно избежать (потому что оба текста находятся на одной линии), но стоит попытаться это сделать. С другой стороны, вы теряете активное пространство между текстами, которое помогает пользователям понять взаимоотношение между двумя отрывками информации.
- В стиле ячейки со значением 2 сокращенный текст притупляет четкость вертикальной линии белого пространства между ним и дополнительным текстом. Когда ширина белого пространства непостоянна от строки к строке, для пользователей становится сложнее просмотреть информацию в дополнительном тексте.

Возможно, вы сможете избежать сокращения текста путем увеличения высоты строки таблицы, чтобы обеспечить перенос текста, но это может быть проблематично по нескольким причинам:

- Вы должны программно проверить длину текста и решить, может ли быть перенос. Вы должны сделать это определение для обеих ориентаций (горизонтальной и вертикальной), потому что ширина таблицы влияет на перенос текста.
- Избегайте ситуации, когда перенос слов в тексте, сделанный для одной ориентации, остается неизменным в другой ориентации.
- Различная высота строк может негативно сказаться на общем виде окна таблицы в вашем приложении вне зависимости от стиля окна таблицы.

Если вы хотите представить строки вашей таблицы в нестандартном виде, то лучше создать уникальный стиль окна таблицы, чем изменять стандартный. Чтобы узнать о том, как создать свои собственные ячейки, смотрите “Customizing Cells” in Table View Programming Guide for iOS.

Text View (Текстовое Окно)

Текстовое окно принимает и отображает множество текстовых строк.



Чтобы узнать больше об описании текстового окна в вашем коде, смотрите [UITextView Class Reference](#).

Внешний вид и Поведение

Текстовое поле - это прямоугольник с закругленными углами, который может быть любой высоты. Текстовое поле поддерживает прокрутку, когда объем контента слишком большой, чтобы уместиться в рамках его границ.

Если текстовое поле поддерживает редактирование пользователем, то, когда пользователь нажимает внутри текстового поля, появляется клавиатура. Метод ввода и шаблон клавиатуры определяется пользовательскими настройками языка. Когда пользователи нажимают на кнопку, помеченную как ".?123" - клавиатура изменяется на раскладку с цифрами, знаками пунктуации и несколькими общими символами.

Руководство

В текстовом поле у вас есть контроль над шрифтом, цветом и выравниванием текста, но только если это применяется ко всему тексту. Иными словами, вы не можете изменить какие-либо свойства только для части текста. Шрифт по умолчанию - это системный шрифт и по умолчанию цвет шрифта - черный, потому что он наиболее читабелен. По умолчанию выравнивание настроено по левой стороне (вы можете изменить и выровнять по центру или правой стороне).

Если вам необходимо использовать различные шрифты, цвета или выравнивания для текстового поля, вместо текстового поля вы можете использовать web-поле и придать стиль тексту, используя HTML.

Вы также можете определить различные стили клавиатуры в зависимости от типа контента, который должен быть введен пользователем. Для изучения описания стилей, которые вы можете использовать, смотрите "Text Field" (страница 134)

Web-view (Веб-окно)

Web-окно - это область которая может отображать различный HTML контент (здесь показано между панелью навигации и панелью инструментов в Mail-приложении на iPhone).



Чтобы узнать больше об описании web-окна в вашем коде, смотрите [UIWebView Class Reference](#).

Внешний вид и Поведение

В дополнение к отображению web-контента, web-окно предоставляет несколько автоматических процессов с web-контентом, таких как конвертация телефонного номера в телефонную ссылку.

Руководство

Если у вас есть web-страница или web-приложение, вы, скорее всего, решите использовать web-окно, чтобы снабдить им простое iOS-приложение, которое предоставляет "обертку" вашей web-страницы или web-приложению. Если вы планируете использовать web-окно для доступа к web-контенту, который вы контролируете, убедитесь, что до этого вы прочитали [Safari Web Content Guide](#).

Избегайте использование web-окна для создания приложения, которое выглядит и действует как небольшой web-браузер. Пользователи ожидают использовать Safari браузер в iOS для отображения веб-контента, и дублирование данных функций в вашем приложении не рекомендуется.

Container View Controller (Контроллер Окна Содержимого)

Container View Controller управляет и представляет набор дочерних окон (или контроллеров окна) в индивидуализированном пути. Примеры определенных системой контроллеров окна содержимого являются tab bar view controller, navigation view controller и split view controller (Вы можете узнать больше об этих компонентах в “Tab Bar” (стр. 105), “Navigation Bar” (стр. 101) и “Split View (iPad Only)” (стр. 112)).

Чтобы узнать больше об описании индивидуализированных контроллеров окна содержимого в вашем коде, см. раздел UINavigationController Class Reference.

Внешний вид и Поведение

Как и следовало ожидать, индивидуализированный контроллер окна содержимого не имеет предопределенного внешнего вида или поведения. При использовании подкласса UINavigationController для создания уникального объекта container view controller вы решаете, сколько контроллеров дочерних окон он содержит, и как они должны быть представлены.

Руководство

Вы можете использовать контроллер окна содержимого для представления контента, благодаря которому пользователи перемещаются по индивидуализированному пути.

Спросите себя, является ли уникальный контроллер окна содержимого действительно необходимым. Пользователей устраивает внешний вид и поведение стандартных контроллеров окна содержимого, таких как split view и tab bar view. Вы должны быть уверены, что потенциальные преимущества уникальных контроллеров окна содержимого перевешивают тот факт, что пользователи не смогут распознать его или сразу знать, как он работает.

Убедитесь, что ваш уникальный контроллер окна содержимого работает в обеих ориентациях. Вам нужно создать контроллер окна содержимого, который предоставляет пользователям согласованную работу, как в горизонтальной, так и в портретной ориентации.

В общем, избегайте кричащих переходов между окнами. При использовании раскадровки для разработки контроллера окна содержимого легко определить уникальные анимации для переходов между содержанием окон. Но в большинстве случаев, целью ярких переходов является отвлечение внимания пользователей от их целей, и это часто уменьшает эстетическую привлекательность вашего приложения.

Alerts, Action Sheets, and Modal Views (Предупреждения, Списки Действий и Модальные Окна)

предупреждения, Списки Действий, предупреждения Модальных Окон являются временными окнами, которые появляются, когда что-то требует пользовательского внимания или когда необходимо предложить дополнительный выбор или функциональность. Пользователи не могут взаимодействовать с приложением в тот момент, когда эти окна отображены на экране.

Чтобы узнать больше об этих типах окон в вашем коде, смотрите “Modal View Controllers” in View Programming Guide for iOS.

Предупреждения

Предупреждение дает пользователям важную информацию, которая приводит к определенному эффекту при использовании приложения (или устройства).



Чтобы узнать больше об использовании предупреждения в вашем коде, смотрите UIAlertView Class Reference.

Внешний вид и Поведение

Предупреждение всплывает в середине экрана приложения и располагается поверх экрана. Неприкрепленное появление предупреждения придает значение тому факту, что его появление обусловлено некоторыми изменениями в приложении или устройстве и не обязательно как результат наиболее частых действий пользователя. Пользователь должен закрыть предупреждение, прежде чем он сможет продолжать использовать текущее приложение.

Предупреждение всегда содержит как минимум одну кнопку, на которую должен нажать пользователь, чтобы закрыть предупреждение. Предупреждение всегда отображает заголовок, а также сообщение, которое предоставляет дополнительную информацию. Внешний вид фона предупреждения определен системой и не может быть изменен.

Примечание: Локальное или Push уведомление также может использовать предупреждение, чтобы осуществить коммуникацию с пользователями. Чтобы узнать больше о локальных и Push уведомлениях, смотри “Notification Center” (стр. 72).

Руководство

Редкость, с которой появляются предупреждения, помогают пользователям воспринимать их серьезно. Убедитесь, что в вашем приложении минимизировано количество предупреждений, а также убедитесь, что каждое из них предоставляет действительно критическую информацию и предлагает полезный выбор.

Избегайте создания предупреждений без необходимости.

Данные предупреждения обычно бесполезны если они:

- Только увеличивают видимость некоторой информации, особенно информации, которая соотносится со стандартными функциями вашего приложения.

Вместо этого вы должны сделать дизайн в привлекательном виде, чтобы отобразить информацию, которая гармонирует со стилем вашего приложения.

- Обновляют информацию по задаче, которая находится в процессе работы (и работает нормально).

Вместо этого используйте окно прогресса или индикатор активности, чтобы предоставлять прогресс, относящийся к ответу пользователя (эти методы ответа описаны в “Progress View” (стр. 141) and “Activity Indicator” (стр. 135)).

- Запрашивают информацию о действиях, инициированных пользователем.

Чтобы получить информацию о действии, которое было инициировано пользователем, даже потенциально рискованное действие, такое как удаление контакта, вы должны использовать Список Действий.

- Информируют пользователей об ошибках или проблемах, с которыми они ничего не могут сделать.

Хотя, может быть, необходимо использовать предупреждение, чтобы сообщить пользователям о критической проблеме, которую они не могут исправить, нужно интегрировать данную информацию в UI, если это возможно. Например, вместо того, чтобы сообщать пользователям каждый раз, что соединение с сервером прервано, отображайте время последнего успешного соединения с сервером.

Вы можете определить текст необходимого заголовка и опционально сообщение, количество кнопок и содержание кнопок предупреждения. Вы не можете индивидуализировать ширину или фон предупреждения, или выравнивание текста (он выровнен по центру).

Как только вы прочтете руководство по дизайну текста предупреждения, вам также необходимо знать следующее:

- **Стиль заголовка с большими буквами** означает, что каждое слово написано с большой буквы, кроме артиклей, союзов и предлогов, которые состоят из четырех или менее букв и когда они не стоят в начале предложения.

- **Стиль заголовка "Предложение"** означает, что первое слово написано с большой буквы, а остальные слова с маленькой, если только это не специфические существительные или прилагательные.

Пишите текст предупреждения, который кратко описывает ситуацию и объясняет, что пользователи должны с ним делать. В идеале текст, который вы пишете, дает пользователям достаточно контекста, чтобы понять, по какой причине появилось предупреждение и решить, какую кнопку нажать.

Старайтесь, чтобы заголовок был достаточно коротким, чтобы уместиться в одну строку, если это возможно. Длинный заголовок предупреждения – слишком сложный, чтобы пользователь мог быстро его прочитать, поэтому следует его сократить или добавить прокрутку в сообщение предупреждения.



Избегайте односложных заголовков, которые не предоставляют никакой полезной информации, таких как "Error" (Ошибка) или "Warning" (Внимание).

По возможности используйте часть предложения. Короткое информативное утверждение часто гораздо проще для понимания, чем полное предложение.

Не сомневайтесь в предоставлении чего-то негативного. Люди понимают, что большинство предупреждений сообщают им о проблемах или предупреждают об опасных ситуациях. Лучше быть негативным и прямым, чем позитивным, но неясным.

Избегайте использования слов "you," "your," "me," and "my" (ваш, ваше, мое, мои и т.д.) как только это возможно. Иногда тексты, которые напрямую обращаются к пользователю, могут быть двусмысленными и даже могут быть интерпретированы как оскорбление.

Используйте заглавные буквы и знаки пунктуации только там, где это необходимо.

Используйте стиль заголовка большими буквами и не расставляйте знаки пунктуации в конце, если заголовок является отрывком фразы или состоит из одного предложения, которое не является вопросом.

Если заголовок состоит из одного предложения, которое является вопросом, используйте стиль заголовка "предложение" и в конце поставьте вопросительный знак. В общем, используйте вопрос для заголовка предупреждения, если это позволит вам избежать добавления сообщения.

Если заголовок состоит из двух и более предложений, используйте стиль заголовка "предложение" и в конце каждого предложения поставьте необходимые знаки пунктуации. Заголовок, состоящий из двух предложений - это очень редкое явление, хотя, вы, возможно, захотите его применить, если это позволит вам избежать добавления сообщения.

Если вы предоставляете опциональное сообщение предупреждения, создайте короткое, законченное сообщение. Используйте стиль "предложение" и соответствующую пунктуацию в конце.



Избегайте создания слишком длинного сообщения для предупреждения. Если это возможно, создавайте сообщение достаточно коротким, чтобы оно отображалось в одну или две строки. Если сообщение слишком длинное, будет необходимо использовать прокрутку, что не является хорошим опытом для пользователя.



Избегайте подсветки текста в предупреждении с описанием, какую кнопку необходимо нажать, например как здесь: "Tap View to see the information" ("Нажмите View, чтобы посмотреть информацию"). В идеале, комбинация недвусмысленного текста предупреждения и логического названия кнопки дает пользователям достаточно информации, чтобы разобраться в ситуации и сделать свой выбор. Однако, если вам необходимо предоставить детальное руководство, соблюдайте нижеследующие инструкции:

- Убедитесь, что вы используете слово "tap" ("нажимать") (а не "touch" or "click" or "choose"), чтобы описать выбор действия.
- Не заключайте заголовок кнопки в кавычки, но сохраняйте написание его с большой буквы.

Убедитесь, что вы проверили внешний вид вашего предупреждения в обеих ориентациях. Потому что высота предупреждения сдерживается в горизонтальной ориентации, и предупреждение может выглядеть немного по-другому, иначе, чем в вертикальной ориентации. Мы рекомендуем, чтобы вы оптимизировали длину текста предупреждения, чтобы он хорошо выглядел (и избегал прокрутки) в обеих ориентациях.

В основном используется двухкнопочное предупреждение. Двухкнопочное предупреждение, обычно является наиболее полезным, потому что это наиболее простой способ для людей выбрать из двух альтернатив. Не очень хорошая идея отображать предупреждение с одной кнопкой, потому что такой вариант только информативен; он не дает пользователям какого-либо контроля ситуации. Предупреждение, которое состоит из 3-х или более кнопок, является значительно более сложным, чем двухкнопочное предупреждение, и его использование следует по возможности избегать. В действительности, если вам необходимо предложить пользователям выбор из более чем двух пунктов, вы должны рассмотреть возможность

использования экшн-списка вместо предупреждения (чтобы узнать больше об использовании списка действий, смотрите “Action Sheet” (стр. 129)).

Используйте должным образом цвета кнопок предупреждения. Кнопки предупреждения подсвечены либо темным цветом, либо светлым. В предупреждении с двумя кнопками, та кнопка, которая располагается слева - всегда темного цвета, а кнопка, которая располагается справа - всегда светлого цвета. В однокнопочном предупреждении, кнопка всегда светлого цвета.

- В двухкнопочном предупреждении, которое предоставляет потенциально рискованное действие, кнопка, которая отменяет действие, должна располагаться справа (и быть более светлой).

- В двухкнопочном предупреждении, которое предоставляет начало действия, необходимое пользователям, кнопка, которая отменяет действие, должна располагаться слева (и быть более темной).

Примечание: Кнопка Cancel (Отмена) может быть либо более светлой, либо более темной и может располагаться как слева, так и справа, в зависимости от того, является ли альтернативный выбор пагубным. Убедитесь, что вы должным образом определили, какая из кнопок в вашем коде является кнопкой Cancel (Отмена) (для более подробной информации смотрите UIAlertView Class Reference).

Нажатие на кнопку Done, в то время как видно предупреждение, должно привести к выходу из приложения, как и ожидается. Это также должно быть идентично нажатию на кнопку Cancel - таким образом, предупреждение отменяется, и действие не подготовлено.

Давайте кнопкам предупреждения короткие, логические заголовки. Наилучший заголовок состоит из одного или двух слов, которые несут смысл, применительно к контексту сообщения. Соблюдайте нижеследующие установки при создании заголовков для кнопок предупреждения:

- Как и для всех заголовков к кнопкам, используйте большие буквы и не используйте знаки пунктуации.

- Используйте глаголы и отглагольные выражения, такие как "Cancel" (Отмена), "Allow" (Позволить), "Reply" (Ответить) или "Ignore" (Игнорировать), которые прямо соотносятся к тексту предупреждения.

- Используйте "OK" как наиболее простую форму принятия чего-либо если нет никакой другой альтернативы. Избегайте использования "Yes" или "No".

- Старайтесь избегать использования таких слов как: “you,” “your,” “me,” и “my” (твой, ваше, мое, мои и т.д.).

Заголовки кнопок, которые используют данные слова, часто являются двусмысленными и снисходительными.

Action Sheet (Список Действий)

Список Действий (как показано ниже на iPhone) отображает набор выборов, соотносящихся с задачами, которые стоят перед пользователем.



Чтобы узнать, как описать список действий (Action Sheet) в вашем коде, смотрите [UIAlertSheet Class Reference](#).

Внешний вид и Поведение

У списка действий есть два основных вида. На iPhone список действий всегда всплывает снизу экрана и перекрывает окна приложения. Стороны списка действий прикреплены к сторонам экрана, что, в свою очередь, укрепляет его связь с приложением и с наиболее частыми действиями пользователя.

На iPad список действий всегда отображается внутри поповера; он никогда не отображается по всей ширине экрана. Список действий может специально открыть поповер, либо он может появиться внутри поповера, который уже открыт. В обоих случаях есть сильная визуальная связь между списком действий и действием пользователя. (Чтобы узнать больше о поповерах, смотри "Popover (iPad Only)" (страница 103).)



Список действий всегда содержит как минимум две кнопки, которые позволяют пользователям выбрать, как завершить их задачу. Когда пользователь нажимает на кнопку, список действий исчезает. Список действий не содержит в себе заголовка или пояснительного текста, потому что он появляется как быстрый ответ на действие пользователя.

Руководство

Используйте список действий, чтобы:

- Предоставить альтернативные пути решения задачи. Список действий позволяет вам предоставить широкий выбор действий, которые имеют смысл в контексте текущей задачи, без необходимости располагать этот выбор на постоянном месте внутри пользовательского интерфейса.
- Предоставить подтверждение, прежде чем завершить потенциально опасную задачу. Список действий заставляет пользователей обдумать потенциально опасный эффект, который может быть произведен и дает им несколько альтернативных путей. Данный способ взаимодействия очень важен для устройств, базирующихся на iOS, потому что иногда пользователи нажимают на контроллеры, совсем этого не желая (случайно).

На iPhone координируйте вид фона списка действий с панелью навигации и панелью инструментов. Используйте черный полупрозрачный фон, если в вашем приложении используется черный цвет для панелей навигации и инструментов. Используйте по умолчанию голубой фон, если ваши панели голубого цвета (по умолчанию).

Все списки действий в вашем iPhone-приложении должны иметь один и тот же цвет фона.

На iPhone включите кнопку Cancel (отмена) - и таким образом пользователи смогут легко и безопасно отменить задачу. Расположите кнопку Cancel внизу списка действий, чтобы вынудить пользователей пересмотреть все варианты альтернативы перед выбором. По умолчанию внешний вид кнопки Cancel соотносится с фоном Списка Действий.

На iPad выберите, отображать ли Список Действий с анимацией или без анимации.

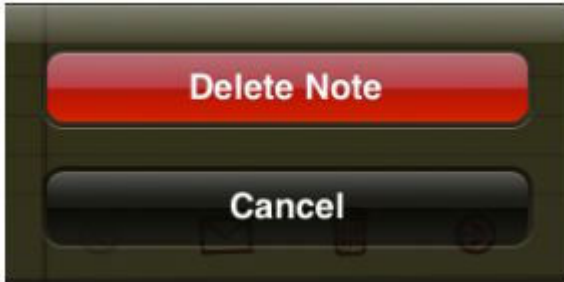
- Отображайте Список Действий без анимации, чтобы предоставить альтернативный выбор относительно задачи, которую пользователь инициализирует за пределами поповера. Без анимации список действий и его поповер появляются одновременно. Когда вы отображаете Список Действий таким образом, стрелки поповера указывают на контроллеры или область, на которую нажал пользователь, инициировав тем самым задачу.

Не вставляете кнопку Cancel, когда список действий отображается без анимации, потому что пользователи могут нажать за пределы поповера, чтобы отменить список действий без выбора одной или более альтернативы.

- Отображайте Список Действий с анимацией, чтобы предоставить альтернативы, относящиеся к задаче, которую пользователь инициирует внутри открытого поповера. С анимацией список действий скользит над открытым контентом поповера.

Анимированный список действий должен содержать кнопку Cancel, потому что пользователи должны иметь возможность отменить список действий без закрытия попувера.

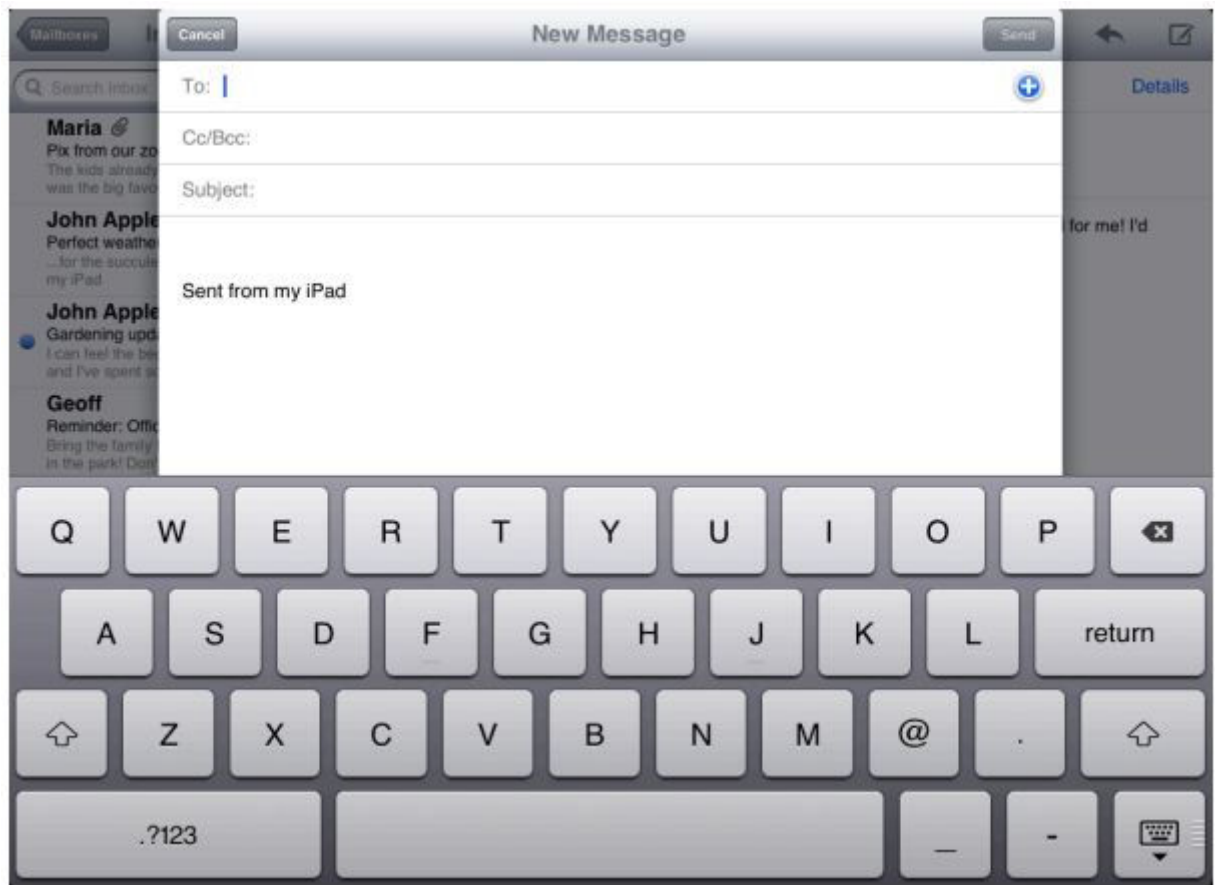
На обоих устройствах используйте красный цвет кнопки, если вам необходимо предоставить кнопку, которая представляет потенциально деструктивное действие. Отображайте красную кнопку сверху списка действий – чем ближе к верхней части списка действий располагается кнопка, тем больше она бросается в глаза. И на iPhone чем дальше деструктивная кнопка находится от низа списка действий, тем меньше есть вероятность, что пользователь нажмет на нее вместо кнопки Home.



Избегайте создания прокрутки в списке действий. Если вы расположите на списке действий слишком много кнопок, пользователем нужно будет прокручивать, чтобы увидеть все действия. Это сбивает с толку пользователей, потому что они должны тратить дополнительное время на обдумывание каждого выбора. Также, пользователям будет очень сложно прокрутить без того, чтобы случайно не нажать не на ту кнопку.

Modal View (Модальное Окно)

Модальное окно - это окно, которое предоставляет замкнутую функциональность в контексте текущей задачи или рабочего процесса.



Чтобы узнать больше про описание модального окна в вашем коде, смотрите [UIViewViewController Class Reference](#).

Внешний вид и Поведение

Модальное окно покрывает весь экран приложения и усиливает пользовательское ощущение ввода отдельного, кратковременного режима, в котором пользователи могут совершить какие-либо действия. На iPad модальные окна, также могут занимать всю площадь одного из родительских окон, таких как поповер.

Модальное окно может отображать, по необходимости, текст и содержать контроллеры, необходимые для предоставления задачи. Модальное окно обычно отображает кнопку, которая завершает задачу и закрывает окно и, при нажатии на кнопку Cancel пользователи могут отказаться от задачи.

Руководство

Используйте модальное окно там, где это необходимо, чтобы предложить возможность выполнить замкнутую задачу, которая относится к основной функции вашего приложения. Модальное окно специально предназначено для мультиэтапной подзадачи, для которой необходимы UI-элементы, которые не относятся к основным элементам пользовательского интерфейса приложения все остальное время.

На iPad выберите стиль модального окна, который подходит к текущей задаче и общему стилю вашего приложения. Вы можете использовать любой из приведенных здесь стилей:

- **Полный экран (Full screen).** Перекрывает весь экран. Этот стиль хорошо подходит для передачи потенциально сложной задачи, которую пользователи могут завершить в контексте модального окна. Например, iPod-приложение использует этот стиль для создания Genius-плейлиста.

- **Список на странице (Page sheet).** Имеет фиксированную ширину в 768 пикселей; высота списка - это текущая высота экрана. В вертикальной ориентации модальное окно покрывает весь экран; в горизонтальной, экран, как это видно по обеим сторонам от модального окна, становится тусклым, чтобы предотвратить взаимодействие с пользователем. Например, Mail (приложение) использует данный стиль для своей задачи по предоставлению сообщений.

- **Лист Формы (Form sheet).** Имеет фиксированные размеры - 540x620 пикселей и отцентрирован на экране. Та область экрана, которая видна за пределами модального окна, становится тусклой, чтобы предотвратить взаимодействие с пользователем. Когда видна клавиатура в горизонтальной ориентации, лист формы передвигается немного ниже статус-панели. Этот стиль хорошо подходит для сбора структурированной информации для пользователя.

- **Текущий контекст (Current context).** Используйте тот же размер, что и для родительского окна. Этот стиль хорошо подходит для отображения модального окна внутри сплит-панели, поповера или другого окна, которое работает не на полный экран.

На iPad не отображайте модальное окно поверх поповера. За исключением предупреждения, ничего не должно отображаться поверх поповера. В крайних случаях, когда вам необходимо отобразить модальное окно как результат действия, которое произвел пользователь на поповере, закройте поповер прежде, чем вы откроете модальное окно.

На iPhone соотносите общий вид модального окна с внешним видом вашего приложения. Например, модальное окно очень часто включает панель навигации, которая, в свою очередь, содержит заголовок и кнопки, с помощью которых можно отменить или завершить задачу в модальном окне. В этом случае, панель навигации, должна иметь точно такой же внешний вид, как и панель навигации в приложении.

На обоих устройствах отображайте заголовок, который идентифицирует задачу, если это необходимо. Вы также можете отображать текст и в других областях окна, который будет более подробно описывать задачу или предоставлять какое-либо руководство. Например, приложение Messages на iPhone предоставляет модальное окно, когда пользователи хотят составить текстовое сообщение. Это модальное окно отображает панель навигации с тем же самым фоном, как и панель навигации всего приложения, а так же заголовок New Message.



На обоих устройствах выберите подходящий стиль перехода для открытия модального окна. Используйте стиль, который соотносится с вашим приложением и увеличивает осознание пользователем временного контекста, который предоставляет модальное окно. Чтобы сделать это, вы можете определить один из представленных стилей перехода:

- **Вертикальный.** Модальное окно выходит вверх из нижнего края экрана и при закрытии, заходит обратно вниз. (Это стиль перехода, который стоит по умолчанию)
- **Переворот.** Текущее окно переворачивается по горизонтали справа налево, чтобы открыть модальное окно. Визуально модальное окно выглядит так, как будто бы оно находится с обратной стороны текущего окна. Когда модальное окно закрывается, оно переворачивается по горизонтали слева направо, открывая предыдущее окно.
- **Частичное перелистывание.** Стиль частичного перелистывания точно такой же, который используется в приложении Maps на iPhone. Визуально это выглядит так, что один угол текущего окна загибается, чтобы открыть модальное окно под ним. Когда пользователь покидает модальное окно, текущее окно заворачивается в изначальную позицию. Вы можете использовать этот стиль, когда модальное окно, которое вы открываете, не особо большое и нет необходимости большего взаимодействия с пользователем, такое как в окне, которое охватывает опции конфигурации.

Примечание: из модального окна, которое было открыто с помощью частичного перелистывания, не может быть открыто другое модальное окно.

Если вы решите варьировать стили переходов модальных окон в вашем приложении, избегайте делать их только ради того, чтобы разнообразить количество переходов. Пользователи сразу подмечают различия переходов и могут заподозрить, что это что-то означает. По этой причине лучше всего установить логическую и четкую схему, которую пользователи могут легко заметить и запомнить, и избежать изменения стилей переходов без причины.

Controllers (Контроллеры)

Контроллеры являются элементами UI, с которыми взаимодействуют пользователи, чтобы выполнить действие или просмотреть информацию. iOS предоставляет большое количество контроллеров, которое вы можете увидеть в вашем приложении.

Activity Indicator (Индикатор Активности)

Индикатор активности показывает, что задача или процесс находятся в действии (здесь показано с ярлыком).



Чтобы узнать больше об описании индикатора активности в вашем коде, смотрите [UIActivityIndicatorView Class Reference](#).

Внешний вид и Поведение

Индикатор активности вращается в то время, как задача находится в процессе, и исчезает, когда задача завершается. Пользователи не взаимодействуют с индикатором активности. По умолчанию индикатор активности - белый.

Руководство

Используйте индикатор активности в панели инструментов или на основном окне, чтобы показать, как происходит процесс, и четко понимать, когда он закончится.

Не отображайте неподвижный индикатор активности, потому что очень часто пользователи соотносят это с тем, что процесс "завис".

Используйте индикатор активности, когда это более необходимо, для того, чтобы уверить пользователей, что их задачи и процессы не "зависли" и было понятно, когда завершится процесс.

Если это необходимо, индивидуализируйте размер и цвет индикатора активности, чтобы он соотносился с фоном окна, на котором он появляется.

Date and Time Picker (Пикер Даты и Времени)

Пикер даты и времени отображает компоненты даты и времени, такие как часы, минуты, дни и года.



Чтобы узнать больше об описании пикера даты и времени в вашем коде, смотрите [UIDatePicker Class Reference](#).

Внешний вид и Поведение

Пикер даты и времени может иметь до четырех отдельных колес, каждое из которых отображает данные по отдельной категории, такие как месяц или час. Пользователи прокручивают каждое колесо до тех пор, пока на нем не отобразятся необходимые данные под планкой выбора, которая проходит через середину пикера. Финальные данные включают в себя данные, отображенные на каждом колесе.

Общий размер пикера даты и времени зафиксирован в том же самом размере, как и размер клавиатуры в iPhone.

У пикера даты и времени есть четыре режима, каждый из которых отображает различное количество колес, которые содержат набор различных значений.

- **Дата и время.** Режим даты и времени отображает колеса для данных календарной даты, часа и минут с опционально дополнительным колесом определения часа до и после полудня (AM/PM). Этот режим - по умолчанию.

- **Время.** Режим времени отображает колеса для значений часа и минут с опционально дополнительным колесом определения часа до и после полудня (AM/PM).

- **Дата.** Режим даты отображает колеса для значений месяца, дня и года.

- **Таймер отсчета.** Режим таймера отсчета отображает колеса со значениями для часа и минут. Вы можете определить общую длительность отсчета до максимум 23 часов и 59 минут.

Руководство

Используйте пикер даты и времени, чтобы позволить пользователям выбирать (вместо того, чтобы набирать текст вручную) данные по дате и времени, которые состоят из различных частей, таких как день, месяц и год. Пикер даты и времени легок в использовании, потому что данные в каждой из его частей имеют относительно малые значения и пользователи уже знают их.

Если это имеет смысл для вашего приложения, измените интервал на колесе, отображающем минуты. По умолчанию минутное колесо отображает 60 значений (от 0 до 59). Если вам необходимо отобразить грубую вариацию выбора, вы можете

установить минутное колесо, чтобы оно отображало больший минутный интервал, чтобы эти интервалы были одинаковыми вместе и составляли 60. Например, вы можете отобразить 15-и минутные интервалы: 0, 15, 30 и 45.

На iPad предоставляйте пикер даты и времени только внутри попувера. Пикер даты и времени не подходит для основного экрана приложения.

Detail Disclosure Button (Кнопка Раскрытия Деталей)

Кнопка раскрытия деталей открывает дополнительные детали или функционал, относящийся к чему-либо (как показано здесь, внутри окна аннотации к карте).



Чтобы узнать больше как описать кнопку раскрытия деталей (UITableViewCellAccessoryDetailDisclosureButton) в вашем коде, смотрите UITableViewCell Class Reference and UIButton Class Reference.

Внешний вид и Поведение

Пользователь нажимает на кнопку раскрытия деталей, чтобы открыть дополнительную информацию или функционал, относящейся к определенной вещи. Дополнительные детали или функционал раскрываются в отдельном окне.

Когда в окне таблицы присутствует кнопка раскрытия деталей, нажатие где-либо на строке не активирует эту кнопку; вместо этого нажатие выбирает пункт в строке или приводит к действию, которое определено поведением приложения.

Руководство

Обычно вы используете кнопку раскрытия деталей в окне таблицы, чтобы дать пользователям возможность увидеть больше деталей или функционала, относящегося к пункту списка. Однако, вы так же можете использовать этот элемент в других типах окон, чтобы предоставить возможность раскрытия более подробной информации или функционала, относящегося к пункту на этом окне.

Info Button (Кнопка Инфо)

Кнопка Инфо открывает детали приложения, часто позади текущего окна.



Чтобы узнать больше об описании Кнопки Инфо в вашем коде, смотрите UIButton Class Reference.

Внешний вид и Поведение

iOS включает в себя два стиля для кнопки Инфо: темная "i" на светлом фоне и светлая "i" на темном фоне. Кнопка Инфо автоматически быстро вспыхивает, когда пользователь нажимает на нее. При нажатии кнопка Инфо приходит в действие, такое как переворот окна в приложении, чтобы "показать заднюю сторону".

Руководство

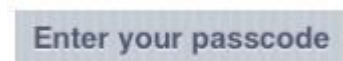
Используйте кнопку Инфо, чтобы открывать детали конфигурации или опций приложения. Вы также можете использовать стиль кнопки Инфо, которая наилучшим образом соотносится с UI вашего приложения.

На iPhone используйте кнопку Инфо, чтобы "перевернуть" экран и открыть больше информации. Обычно обратная сторона экрана отображает опции конфигурации, которые не должны быть на основном UI.

На iPad избегайте использования кнопки Инфо, чтобы "переворачивать" весь экран. Вместо этого вы должны использовать кнопку Инфо, чтобы показать пользователям, что они могут иметь доступ к выпадающему окну, которое содержит информацию или дополнительные детали.

Label (Лейбл)

Лейбл отображает статический текст.



Чтобы узнать больше об описании лейбла в вашем коде, смотрите UILabel Class Reference.

Внешний вид и Поведение

Лейбл отображает любое количество статического текста. Пользователи не взаимодействуют с лейблами, кроме потенциальной возможности скопировать текст.

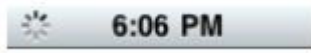
Руководство

Вы можете использовать лейбл, чтобы назвать или описать части вашего UI или чтобы предоставить короткие сообщения пользователю. Лейбл наиболее подходит к отображению относительно малого количества текста.

Позаботьтесь о том, чтобы сделать ваши лейблы четкими. Не жертвуйте четкостью в пользу причудливых шрифтов или ярких цветов.

Индикатор Активности Сети

Индикатор активности сети появляется в статус-панели и показывает, что происходит активность сети (здесь показано слева от времени)



В вашем коде вы используете UIApplication method `networkActivityIndicatorVisible`, чтобы контролировать видимость индикатора.

Внешний вид и Поведение

Индикатор активности сети вращается в статус-панели в то время, когда происходит активность сети. Индикатор исчезает в тот момент, когда активность сети останавливается. Пользователи никак не взаимодействуют с индикатором активности сети.

Руководство

Отображайте индикатор активности сети, чтобы предоставить обратную связь, когда ваше приложение имеет доступ к сети на более чем пару секунд. Если процесс происходит менее чем за этот период времени, вам нет необходимости отображать индикатор активности сети, потому что индикатор исчезнет до тех пор, пока пользователь не определит его присутствие.

Индикатор страницы

Индикатор страницы показывает, сколько окон открыто, и какое из них отображается в текущий момент.



Чтобы узнать больше об описании индикатора страницы в вашем коде, смотрите [UIPageControl Class Reference](#).

Внешний вид и Поведение

Индикатор страницы отображает точку для каждого открытого в настоящий момент окна в приложении. Слева направо точки показывают порядок, в котором окна были открыты (самая левая точка отображает первое окно). Текущее видимое окно идентифицируется сиянием на той точке, которая ее представляет. Пользователи нажимают слева или справа от сияющей точки, чтобы увидеть предыдущее или последующее открытое окно.

Точки индикатора страницы не исчезают и не сжимаются друг в друга, если появляется больше страниц. Экран в вертикальной ориентации может вместить примерно 20 точек; если вы попытаетесь отобразить больше точек, чем может вместиться на экране, они будут обрезаны.

Руководство

Используйте индикатор страницы, когда каждое окно в вашем приложении равнозначно каждому другому окну. Не используйте индикатор страницы, если в вашем приложении отображается информация в иерархии окон, потому что индикатор страницы не помогает пользователям отслеживать свои шаги по специфическому пути.

В вертикальной ориентации разместите индикатор страницы по центру, между нижним краем открытого окна и нижним краем экрана, где он всегда будет заметен. Убедитесь, что вы не пытаетесь отобразить слишком много точек на текущей ориентации экрана.

На iPad найдите пути отображения вашего контента на одном экране. На большом iPad-экране множество параллельных экранов не работает как следует, и, таким образом, необходимость в индикаторе страницы уменьшается.

Picker (Пикер)

Пикер отображает набор данных, из которых пользователь выбирает один пункт.



Чтобы узнать больше об описании пикера в вашем коде, смотрите UIPickerView Class Reference.

Внешний вид и Поведение

Пикер – это сгенерированная версия пикера даты и времени. Так же как и в пикере даты и времени, пользователи крутят колесо (или колеса) пикера, пока не появятся данные, которые им нужны. Общий размер пикера, включая его фон, фиксирован в том же размере, как и клавиатура на iPhone. (Для более детальной информации о пикере даты и времени смотрите “Date and Time Picker” (стр 125).

Руководство

Используйте пикер, чтобы сделать возможность пользователям очень просто выбирать из набора значений. Очень часто лучше использовать пикер, когда пользователи очень

хорошо знакомы с общим набором значений. Это потому, что много, если не большинство значений, скрыты, когда колесо закреплено в какой-либо позиции. Если вам необходимо предоставить большой набор выборов, которые не так хорошо знакомы вашим пользователям, пикер - это не самый лучший контроллер для использования.

Подумайте над тем, чтобы использовать окно таблицы вместо пикера, если вам необходимо отобразить большое количество значений. Это потому что большая высота окна таблицы делает возможным просмотр всех данных быстрее.

Используйте полупрозрачную планку секции, чтобы отобразить информацию, относящуюся к контексту, такую как величины измерения. Не отображайте такие надписи над пикером или на самом колесе.

На iPad предоставляйте пикер только внутри поповера. Пикер не подходит к основному экрану (main screen).

Progress View (Окно Прогресса)

Окно прогресса отображает прогресс задачи или процесса, который имеет известную длительность (здесь показано с текстом).



Чтобы узнать больше об описании окна прогресса в вашем коде, смотрите [UIProgressView Class Reference](#).

Внешний вид и Поведение

iOS предоставляет два стиля для окна прогресса. Внешний вид каждого из этих стилей одинаков за исключением высоты.

- **Стиль по умолчанию** - белый и имеет некий объем (вес), который хорошо подходит к использованию в основной области контента приложения.

- **Стиль плашки** - тоньше, чем стиль по умолчанию, что хорошо подходит для использования в панель инструментов. Стиль плашки также белый по умолчанию.

С той скоростью, с которой выполняется задача или процесс, анимация прогресса заполняется слева направо. В любой момент по пропорции заполненной и не заполненной области в окне прогресса, пользователи смогут определить как быстро завершится задача или процесс. Пользователи не взаимодействуют с окном прогресса.

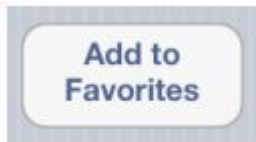
Руководство

Используйте окно прогресса, чтобы предоставлять обратную связь пользователям по задаче, длительность действия которой хорошо известна, особенно когда очень важно показать примерную длительность задачи. Когда вы отображаете окно прогресса, вы сообщаете пользователям, что их задача находится в процессе подготовки, и предоставляете им достаточно информации, чтобы решить, хотят ли они ждать завершения задачи или захотят ли они ее отменить.

При необходимости индивидуализируйте внешний вид окна прогресса с целью приведения в соответствие со стилем вашего приложения. Вы можете указать цвет или изображение для обоих “track” и “fill” окон прогресса.

Rounded Rectangle Button (Прямоугольная кнопка с закругленными краями)

Прямоугольная кнопка с закругленными краями предоставляет специфическое действие приложения.



Чтобы узнать больше об описании прямоугольной кнопки с закругленными краями в вашем коде, смотрите UIButton Class Reference.

Внешний вид и Поведение

Прямоугольная кнопка с закругленными краями имеет радиус угла такой же, как и радиус угла сгруппированного окна таблицы. По умолчанию фон кнопки - белый.

Руководство

Используйте прямоугольную кнопку с закругленными краями, чтобы инициировать действие. Когда вы предоставляете заголовок для прямоугольной кнопки с закругленными краями, следуйте следующим принципам:

- Используйте стиль заголовка большими буквами (что означает писать большими буквами каждое слово, кроме артиклей, соединительных союзов и предлогов из четырех или менее букв).
- Избегайте создания слишком длинных заголовков. Слишком длинный текст должен быть сокращен, что адаптирует его для понимания пользователем.

Вы можете определить заголовок или изображение, чтобы отобразить в прямоугольной кнопке с закругленными углами. (В качестве некоторого руководства по обеспечению масштабируемого размера фонового изображения, см. раздел “Tips for Creating Resizable Images” (стр. 160)). Вы также можете определить, подсвечивание кнопки при нажатии на нее, и вид заголовка, когда кнопка подсвечена.

Scope bar (Панель Возможностей)

Scope bar предоставляет пользователям возможность определять границы поиска. Панель возможностей доступна только во взаимодействии с панелью поиска. Панель возможностей показана здесь, ниже панели поиска.



Чтобы узнать больше об описании панели поиска и панели возможностей в вашем коде, смотрите [UISearchBar Class Reference](#).

Внешний вид и Поведение

Когда присутствует панель поиска, панель возможностей может появиться рядом с ним. Панель возможностей отображается под панелью поиска независимо от ориентации, до тех пор, пока вы не используете контроллер отображения поиска в вашем коде (для более подробной информации по поводу того, как это работает, смотрите [UISearchDisplayController Class Reference](#)). Если вы используете контроллер отображения поиска (search display controller), панель возможностей отображается внутри панели поиска, справа от поля поиска, когда устройство находится в горизонтальной ориентации (в вертикальной ориентации, он находится под панелью поиска).

Панель возможностей содержит кнопки, на которые пользователи могут нажать для выбора границы поиска, и он принимает тот же самый вид, который вы определили для панели поиска.

Руководство

Будет полезно отобразить панель возможностей, когда очень четко определены типичные категории, в которых пользователи могут захотеть проводить поиск. Например, пользователи часто хотят сузить их поиск до одного поля в e-mail-сообщении.

Вы можете индивидуализировать панель возможностей, предоставляя фоновое изображение. Кроме того, вы можете определить различные выступления для включения и отключения состояний кнопок панели возможностей и разделителей между ними.

Search bar (Панель Поиска)

Search bar принимает от пользователей текст, который может быть использован как вводные данные для поиска (показано здесь с текстом-заполнением и кнопкой Bookmarks).



Чтобы узнать, как описать панель поиска в вашем коде, смотрите [UISearchBar Class Reference](#).

Внешний вид и Поведение

Панель поиска выглядит как текстовое поле с закругленными краями. По умолчанию панель поиска отображает иконку поиска с левой стороны. Когда пользователь нажимает на панель, появляется раскладка клавиатуры; когда пользователь закончил набор слова для поиска, эти вводные данные подхватываются в специфическом для приложения виде.

В дополнение панель может отображать несколько опциональных элементов:

- **Текст-заполнение.** Этот текст может определять функцию контроллера (например, "Search") или напоминать пользователям в каком контексте им необходим поиск (например, "YouTube" или "Google").

- **Кнопка Закладки.** Эта кнопка может предоставлять сокращенный путь к информации, которую пользователи захотят очень легко найти снова. Например, кнопка Закладки в режиме поиска в приложении Maps предоставляет доступ к локациям, предыдущим критериям поиска и контактам, которые до этого были добавлены в закладки.

Кнопка Закладки должна быть видимой, только если нет текста, предоставленного пользователем или текста, который не является заполнением в панели поиска, потому что кнопка Clear (Очистить) является видимой, только если в панели поиска находится текст, который пользователь, возможно, захочет стереть.

- **Кнопка Clear (Очистить).** Большинство панелей поиска включают кнопку Clear (Очистить), которая позволяет пользователям удалять содержание панели поиска одним нажатием. (Кнопка Clear (Очистить) представляет собой серый кружок с белым знаком "x" внутри).

Когда внутри панели поиска находится какой-либо текст, который не является заполнением, кнопка Clear (Очистить) становится видимой, таким образом пользователи могут стереть текст. Если внутри панели поиска не находится какого-либо текста, предоставленного пользователем или заполнения, кнопка Clear (Очистить) находится в скрытом состоянии, потому что нет необходимости удалять содержимое панели поиска.

- **Иконка списка результатов.** Эта иконка указывает на наличие результатов поиска. Когда пользователь нажимает на иконку в списке результатов, ваше приложение может отображать результаты их последнего поиска.

- Описывающий заголовок, называемый "**подсказкой**", который появляется над панелью поиска. Например, подсказка может являться короткой фразой, которая предоставляет введение или специфическое для данного приложения содержание для панели поиска.

Руководство

Используйте панель поиска, чтобы дать возможность поиска внутри вашего приложения. Не используйте текстовое поле, потому что у панели поиска нет стандартного внешнего вида, который ожидает увидеть пользователь.

Вы можете индивидуализировать панель поиска, указав уникальный вид фона и предоставления соучаствующих изображений. Для фона можно указать цвет или изображение, или выбрать один из следующих стандартных цветов фона:

- Голубой (Это градиент по умолчанию, который сочетается с внешним видом панели инструментов (toolbar) и панели навигации (navigation bar), в том виде как они выглядят по умолчанию)

- Черный
- Черный прозрачный

Если вы решите предоставить фоновое изображение, хорошо предоставить масштабируемый размер изображения, чтобы узнать больше о создании этого, см. раздел “Tips for Creating Resizable Images” (page 165).

Segmented Control (Сегментированный Контроллер)

Сегментированный контроллер - это набор линейных сегментов, каждый из которых функционирует как кнопка, которая может отображать различные окна (view).



Чтобы узнать больше об описании сегментного контроллера в вашем коде, смотрите [UISegmentedControl Class Reference](#).

Внешний вид и Поведение

Длина сегментного контроллера определяется количеством его сегментов; высота сегментного контроллера является фиксированной. Ширина каждого сегмента является пропорциональной, основываясь на общем количестве сегментов. Когда пользователь нажимает на сегмент, сегмент отображает выбранное состояние.

Руководство

Используйте сегментный контроллер, чтобы предоставлять близкие, но взаимоисключающие варианты выбора.

Убедитесь, что на каждый из сегментов легко нажать. Чтобы поддерживать комфортную зону для нажатия (которая составляет 44x44 пикселя) для каждого сегмента, вам необходимо лимитированное количество сегментов. На iPhone сегментный контроллер должен иметь около 5-и сегментов.

По возможности поддерживайте постоянный размер содержания каждого сегмента. Так как все сегменты в сегментном контроллере имеют одинаковую ширину, то очень плохо будет выглядеть, если в одном сегменте есть содержание, а в других нет.

Избегайте совмещение текста и картинки в односегментном контроллере. Сегментный контроллер может содержать текст или картинку. Отдельный сегмент может содержать либо текст, либо изображение, но не то и другое вместе. Лучше всего избегать расположения текста в одних сегментах и изображений на других сегментах в одном сегментном контроллере.

При необходимости индивидуализируйте внешний вид сегментированного контроллера. Например, можно указать уникальный цвет фона или изображение. Если указать фоновое изображение, вы также можете указать другое фоновое изображение для использования в выбранном виде сегмента, и уникальный внешний вид разделителей между сегментами. (В некоторых случаях, хорошо предоставить изображение масштабируемого размера, чтобы узнать больше о создании этого, см. раздел “Tips for Creating Resizable Images” (стр. 160).)

При использовании уникального сегментированного контроллера, вы должны убедиться, что автоматическое центрирование текста или содержимого изображения по-прежнему выглядит хорошо. При необходимости вы можете использовать bar metrics API для индивидуализации содержимого внутри сегментированного контроллера (чтобы узнать больше об указании панели метрик, см. customization APIs описанной в UISegmentedControl Class Reference).

Slider (Слайдер)

Слайдер позволяет пользователям сделать настройки значений или процесса внутри предела дозволённых величин. (здесь показано с индивидуализированными изображениями слева и справа).



Чтобы узнать больше об описании слайдера в вашем коде, смотрите UISlider Class Reference.

Внешний вид и Поведение

Слайдер состоит из пути (track) и ярлыка (thumb) – круглый контроллер, который может двигать пользователь, и опциональных изображений, которые передают значения сдвига значений вправо или влево. Когда пользователи передвигают ярлык по слайдеру, значение или процесс постоянно обновляется и это передается на пути (track).

Руководство

Используйте слайдер, чтобы дать пользователям систему управления над значениями, которые они могут выбрать или управление текущим процессом.

Если это необходимо, индивидуализируйте внешний вид слайдера. Например, вы можете выбрать что-то из представленного ниже:

- Отобразите слайдер или горизонтально, или вертикально.
- Установите ширину слайдера, чтобы он подходил к общему UI вашего приложения.
- Определите внешний вид ярлычка (thumb), чтобы пользователи могли с легкостью распознавать, активен ли слайдер.
- Предоставьте картинки по обеим сторонам от слайдера, чтобы помочь пользователям понять, как работает слайдер.

Обычно эти уникальные картинки соотносятся с минимальными и максимальными значениями, которые контролирует слайдер. Например, слайдер, который контролирует размер шрифта, может отображать очень маленькую букву на одном конце и очень большую букву на другом конце.

- Определите различный внешний вид дорожки (track) в зависимости от того, где находится ярлычок (thumb) и в каком состоянии находится контроллер.

Stepper (Контроллер пошагового изменения величины)

Stepper увеличивает или уменьшает значение на постоянную величину.



Чтобы узнать больше о stepper в своем коде, см. [UIStepper Class Reference](#).

Внешний вид и Поведение

Stepper состоит из двух сегментов управления, в которых один сегмент показывает знак плюс, а другой сегмент – минус. Пользователи нажимают на элемент, чтобы увеличить или уменьшить значение. Stepper не отображает значение, изменяемое пользователем.

Руководство

В общем, используйте stepper, когда пользователям, возможно, потребует внести незначительные изменения в значении. Например, имеет смысл использовать stepper, чтобы установить количество копий в Printer Options (параметрах печати), так как пользователи редко очень сильно изменяют это значение. С другой стороны, не имело бы смысла использовать stepper, чтобы помочь пользователям выбрать диапазон страниц, поскольку эти значения могут сильно меняться.

Сделайте очевидным влияние stepper на изменения. Stepper не отображает значения, поэтому необходимо убедиться, что пользователи знают, какое значение они меняют, когда они используют stepper.

Переключатель (Switch)

Переключатель предоставляет два взаимно исключающих выбора или состояния (используется только в окне таблицы).



Чтобы узнать больше об описании переключателя в вашем коде, смотрите [UISwitch Class Reference](#).

Внешний вид и Поведение

Переключатель отображает значение, которое в данный момент находится в работе; пользователи переключают контроллер, чтобы выбрать (и открыть) другое значение. Пользователи также могут нажать на контроллер, чтобы переключиться между двумя выборами.

Руководство

Используйте переключатель в строке таблицы, чтобы дать пользователям два простых, диаметрально противоположных выбора, которые определяют состояние чего-либо, такие как да\нет (yes\no) или вкл\выкл (on\off). Используйте предсказуемую пару значений, чтобы пользователям не пришлось переключать контроллер, чтобы выяснить, что означает обратное значение.

Вы можете использовать контроллер переключателя, чтобы изменить состояние других UI-элементов в окне. В зависимости от выбора, который делает пользователь, список новых опций должен появляться или исчезать, или список опций должен становиться активным или неактивным.

При необходимости индивидуализируйте внешний вид переключателя. Вы можете задать тон для открытого состояния, который соответствует виду вашего приложения.

Примечание: Если вы создали уникальный переключатель для возможности отображения уникального внешнего вида, он не будет получать внешний вид iOS5 переключателя.

Text Field (Текстовое Поле)

Текстовое поле принимает одну строку, которую вводит пользователь (здесь показано с описанием и текстом-заполнения).



Чтобы узнать больше об описании текстового поля и его индивидуализации, чтобы отображать изображения и кнопки, смотрите UITextField Class Reference.

Внешний вид и Поведение

Текстовое поле - это фиксированное по высоте поле с закругленными углами. Когда пользователи нажимают на текстовое поле, появляется раскладка клавиатуры; когда пользователь нажимает на клавиатуре кнопку Return (Ввод), текстовое поле удерживает введенный текст в том виде, которое соответствует специфике приложения.

Руководство

Используйте текстовое поле, чтобы получить небольшой объем информации от пользователя. Прежде чем решить использовать текстовое поле, посмотрите, можно ли использовать другие контроллеры, с помощью которых ввод информации может быть намного проще реализован, например, пикер или список.

Индивидуализируйте текстовое поле, если это поможет пользователям понять, как они должны это использовать. Например, вы можете отображать уникальные изображения слева или справа от текстового поля, или добавить кнопку, которая предоставляется системой, такую как кнопка Bookmarks (Закладки). Обычно вы должны

использовать левый конец текстового поля, чтобы определять его назначение и правый конец текстового поля, чтобы определять наличие дополнительных функций, таких как закладки.

Отображайте кнопку Clear, с правой стороны текстового поля, когда это необходимо. Когда этот элемент присутствует, нажатие на него очищает содержание текстового поля вне зависимости от каких-либо других изображений, которые вы можете отображать поверх.

Отобразите совет в текстовом поле, если это поможет пользователям понять его предназначение, например “Name” or “Address”. Текстовое поле может отображать такой текст-заполнение, когда нет другого текста в поле.

Определяйте различные типы клавиатуры, чтобы приспособить ее к различным типам контента, который по вашим ожиданиям должен ввести пользователь.

Например, возможно вы захотите, чтобы пользователям было проще вводить URL, PIN код или телефонный номер. Примечание: помните, что у вас нет контроля над методом ввода с клавиатуры и ее шаблоном, которые определены пользовательскими установками языка. iOS предоставляет несколько различных типов клавиатуры, каждая из которых создана, чтобы содействовать вводу различных типов данных. Чтобы узнать больше о доступных типах клавиатуры, смотрите документацию к `UIKeyboardType`. Чтобы узнать больше об управлении клавиатурой в вашем приложении, прочтите “Managing the Keyboard” in iOS Application Programming Guide.

System-Provided Buttons and Icons (Кнопки и Иконки, предоставленные системой)

Чтобы способствовать последовательному пользовательскому опыту (и сделать вашу работу проще), iOS предоставляет множество стандартных иконок для использования в панели навигации и панели инструментов, и иконки для использования в панели ярлыков.

Вы должны ознакомиться с правилами, которые управляют использованием кнопок и иконок, предоставленных системой, вне зависимости от типа приложения, которое вы делаете. И таким образом вы можете:

- Правильно использовать элементы, которые предоставлены системой
- Избегать дизайна индивидуализированных иконок, которые выглядят слишком похожими на иконки, которые предоставлены системой.

Примечание: кнопки и иконки, которые предоставлены системой, не предоставлены в отдельных файлах. Вместо этого вы используете API имена символов, чтобы определить их, когда вы создаете стандартные UIKit кнопки. Например, символ, который идентифицирует кнопку Действия (находится в таблице 7-2 (стр 136)) - это `UIBarButtonItemSystemItemAction`.

Вы также можете определить эти имена символов в Interface Builder: в Атрибутах контроллера для вашей кнопки выберите иконку из выпадающего меню Identifier. Для более подробной информации смотрите “Setting the Appearance of Bar Button Items” в Interface Builder User Guide.

Если вы не находите кнопку для панели инструментов или панели навигации или панели ярлыков, которая предоставлена системой и имеет подходящее значение для специфической функции в вашем приложении, вы должны сделать уникальный дизайн кнопки или иконки. Для изучения руководства, которое поможет вам сделать это, смотрите “Icons for Navigation Bars, Toolbars, and Tab Bars” (стр. 170).

Standard Buttons for Use in Toolbars and Navigation Bars (Стандартные Кнопки для Использования в Панелях Инструментов и Навигации)

iOS делает доступными множество стандартных иконок, которые пользователи видят в панели инструментов и панели навигации. Эти кнопки показаны в Таблице 7-2 (стр. 150-151) и доступны в двух стилях, каждый из которых подходит для специфического использования, описанного ниже:


- **Окантованный стиль** - например, кнопка Add, в панели навигации в приложении Contacts на iPhone, использует окантованный стиль. Этот стиль подходит как для панели навигации, так и для панели инструментов.







- **Простой стиль** - например, кнопка Compose, в панели инструментов приложения Mail, использует простой стиль. Этот стиль подходит только для панели инструментов. Если вы определите простой стиль кнопки в панели навигации, она автоматически будет переделана в окантованный стиль.

Как и со всеми кнопками, предоставленными системой, вы должны избегать использования кнопок, описанных в Таблице 7-2, чтобы предоставить действия, отличные от тех, для которых они созданы. В частности, избегайте выбора кнопок, основываясь на их внешнем виде, без отсылки к их задокументированному значению. Для уточнения причин, почему очень важно использовать эти иконки правильно, смотрите “Use UI Elements Consistently” (стр. 49).

Чтобы выяснить названия символов и использовать их для определения этих кнопок, смотрите документацию для UIBarButtonItemSystemItem in UIBarButtonItem Class Reference.

Таблица 7-2 Стандартные кнопки, доступные для панели инструментов и панели навигации (простой стиль)







Кнопка	Название	Значение
	Действие (Action)	Открывает список действий, который позволяет пользователям произвести специфическое для приложения действие.

Кнопка	Название	Значение
	Камера (Camera)	Открывает список действий, который отображает фото-пикер в режиме камеры
	Составление (Compose)	Открывает новое окно сообщения в режиме редактирования
	Закладки (Bookmarks)	Показывает специфические закладки приложения
	Поиск (Search)	Отображает поле поиска
	Добавить (Add)	Создает новый пункт
	Мусор (Trash)	Удалить текущий пункт
	Организация (Organize)	Передвинуть или перенаправить пункт внутри приложения, например, в папку
	Ответ (Reply)	Отправить или перенаправить пункт в другое место
	Стоп (Stop)	Остановить текущий процесс или задачу
	Обновить (Refresh)	Обновить контент (используйте только когда это необходимо; либо обновление не происходит автоматически)
	Играть (Play)	Начинает проигрывание медиа или слайдов
	Перемотка Вперед (FastForward)	Быстрая перемотка вперед по проигрывающемуся медиаконтенту или слайдам
	Пауза (Pause)	Приостанавливает проигрывающийся медиаконтент или слайды (при этом контент сохраняется)
	Перемотка Назад (Rewind)	Перемотка назад через медиаконтент или слайды

В дополнение к кнопкам, показанным в Таблице 7-2, вы также можете использовать кнопки Edit, Cancel, Save, Done, Redo, and Undo, показанные в Таблице 7-3, для поддержки редактирования или других типов манипуляций с контентом в вашем приложении.

Чтобы выяснить названия символов для использования этих кнопок, смотрите документацию для `UIBarButtonItem` in `UIBarButtonItem Class Reference`.

Таблица 7-3 Окантованные кнопки действий для использования в панели навигации и панели инструментов

Кнопка	Название	Значение
	Редактирование (Edit)	Вход в режим редактирования или манипуляций с контентом
	Отмена (Cancel)	Выход из режима редактирования или манипуляций с контентом без сохранения изменений
	Сохранить (Save)	Сохранить изменения, и, если это необходимо, выйти из режима редактирования или манипуляции с контентом
	Готово (Done)	Выход из текущего режима и сохранение изменений, если таковые имелись
	Отменить Действие (Undo)	Отменяет самое последние действие
	Сделать Снова (Redo)	Снова выполняет самое последнее действие, которое было отменено

Примечание: В общем, есть не более одной синей кнопки на панели. Синяя кнопка представляет собой действие, которое пользователь, скорее всего, применяет в текущем контексте.

Кнопки, предоставленные в Таблице 7-3, подходят как для панели навигации, так и для панели инструментов, и доступны только в окантованном стиле. Если вы определяете простой стиль для какой-либо из этих кнопок, они будут переведены в окантованный стиль.

В iOS 4 и более, вы можете использовать кнопку перелистывания, предоставленную системой, в панели инструментов (для изучения более подробной информации смотрите документацию `UIBarButtonItemSystemItemPageCurl` in `UIBarButtonItem` Class Reference). Кнопка перелистывания страницы недоступна для использования в панели навигации.



Кнопка перелистывания позволяет вам дать пользователям способ перелистнуть вниз угол экрана, чтобы увидеть информацию под ним. Например, приложение Maps позволяет пользователям приподнимать правый нижний угол окна карты, чтобы получить доступ к кнопкам для управления картой.













Не используйте кнопку перелистывания, чтобы разворачивать экран (переход между экранами). Если вам необходимо перевернуть экран, вместо этого используйте кнопку Инфо (для изучения более подробной информации о кнопке Инфо, смотрите “Info Button” (стр. 138)). Также убедитесь, что некоторые из перелистнутых страниц все еще видны на экране, чтобы подчеркнуть временную природу действия по перевороту экрана.

Если верхняя страница исчезает, переворот страницы начинает слишком сильно напоминать полно-экрановой переход, и пользователи теряют свой контекст.

Standard Icons for Use in Tab Bars (Стандартные Иконки для Использования в Панели Ярлыков)

iOS предоставляет стандартные иконки, описанные в Таблице 7-4, для использования в Tab Bar.

Таблица 7-4 Стандартные иконки для использования в панели ярлыков

Иконка	Название	Значение
	Закладки (Bookmarks)	Показывают специфические для приложения закладки
	Контакты (Contacts)	Показывает контакты
	Загрузки (Downloads)	Показывает загрузки
	Избранное (Favorities)	Показывает то, что определено пользователем как избранное
	Особенное (Featured)	Показывает особый контент для этого приложения
	История (History)	Показывает историю пользовательских действий
	Более (More)	Показывает дополнительные пункты Tab Bar
	Самое Последнее (MoreRecent)	Показывает самые последние пункты
	Самое Просмотренное (MostViewed)	Показывает самое популярное среди всех пользователей
	Последнее (Recents)	Показывает что-то, куда входил пользователь в период, определенный приложением
	Поиск (Search)	Входит в режим поиска
	Самое Рейтинговое (TopRated)	Показывает что-либо наиболее рейтинговое, что было определено пользователем

Как и все стандартные кнопки и иконки, очень важно использовать иконки панели ярлыков в соответствии с их задокументированным значением. В частности позаботьтесь об использовании ваших иконок, основываясь на семантическом значении, а не на внешнем виде. Это поможет пользовательскому интерфейсу вашего приложения иметь смысл, даже если иконки ассоциируются со специфическим




значением, изменяющим их внешний вид. Для изучения причин, почему очень важно правильно использовать эти иконки, смотрите “Use UI Elements Consistently” (стр. 49).

Чтобы выяснить какие названия у символов, определяющих данные иконки, смотрите документацию для `UITabBarItem` in `UITabBarItem Class Reference`.

Standard Buttons for Use in Table Rows and Other UI Elements (Стандартные кнопки для использования в Строках Таблицы и Другие UI-Элементы)

iOS предоставляет кнопки, описанные в Таблице 5-7, для использования в строках таблицы и других элементах.

Таблица 7-5 Стандартные кнопки для использования в строках таблицы и других элементах UI

Кнопка	Название	Значение
	Добавить Контакт (ContactAdd)	Показывает пользователю пикер, чтобы добавить контакт в какой-либо пункт.
	Раскрытие Деталей (DetailDisclosure)	Отображает новое окно, которое содержит дополнительную информацию о текущем пункте.
	Инфо (Info)	Переворот на заднюю сторону окна, чтобы отображать опции конфигурации или более подробную информацию. Примечание: Кнопка Инфо (Info) также доступна со светлой буквой "i" в темном круге.

Эти кнопки могут быть использованы согласно их определенному значению, также как и все остальные стандартные кнопки и иконки. Другими словами, избегайте выбора кнопки, основываясь на ее внешнем виде, без соотношения с задокументированным значением. Для изучения причин, почему очень важно использовать эти кнопки правильно, смотрите “Use UI Elements Consistently” (стр. 49).

Хотя кнопка раскрытия информации обычно используется в строках таблицы, она может быть использована где угодно. Для более подробной информации об этой кнопке, смотрите “Detail Disclosure Button” (стр. 137). iOS также предоставляет набор контроллеров для использования только в строках таблицы; для более подробной информации, смотрите “Table View” (стр. 113).

Для изучения более подробной информации об использовании кнопок Добавить Контакт (ContactAdd) и Инфо (Info) в вашем приложении смотрите документацию для `UIButtonType` in `UIButton Class Reference`. Для изучения информации по использованию кнопки Раскрытия Деталей (DetailDisclosure) в вашем приложении смотрите `UITableViewCellAccessoryDetailDisclosureButton` in `UITableViewCell Class Reference`.

Custom Icon and Image Creation Guidelines

Руководство по Созданию Кастомных Иконок и Изображений

Каждому приложению необходимо иметь иконку и картинку запуска. Рекомендуется, чтобы приложения также предоставляли иконку для iOS для отображения в результатах поиска Spotlight (и, если это необходимо, в Настройках) В дополнение некоторым приложениям необходимы индивидуализированные иконки, которые представляют индивидуализированные типы документов или специфические функции приложения и режимы в панелях Навигации, Инструментов и Ярлыков.

В отличие от остальных индивидуальных работ в вашем приложении, эти иконки и изображения должны соответствовать специфическим критериям, чтобы iOS могла отображать их должным образом. В дополнение файлы иконки и изображения имеют определенные требования к названию. Таблица 8-1 содержит информацию об этих иконках и изображениях и предоставляет ссылки на специфические руководства по их созданию. Чтобы узнать, как называть эти файлы и как описывать их в вашем коде, смотрите “Application Icons” in iOS Application Programming Guide и “Application Launch Images” in iOS Application Programming Guide.

Вы можете индивидуализировать многие элементы пользовательского интерфейса, предоставляя изменяемый размер изображения для элементов фона. Чтобы узнать, как создать изменяемый размер изображения, который хорошо выглядит и хорошо работает, см. раздел “Tips for Creating Resizable Images” (стр.160).

Примечание: Чтобы поддержать автономность разрешения, вы должны предоставить версии ваших иконок и изображений в высоком разрешении в дополнение к тем ресурсам, которые вы уже предоставили. Для ознакомления с руководством по созданию работ в высоком разрешении, смотрите “Tips for Creating Great Artwork for the Retina Display” (стр. 158).

Таблица 8-1 Индивидуализированные иконки и изображения

Описание	Размер для iPhone и iPod touch (в пикселях)	Размер для iPad (в пикселях)	Руководство
Иконка приложения (обязательно)	57 x 57 114 x 114 (высокое разрешение)	72 x 72 114x114 (высокое разрешение)	“Application Icons” (страница 166)
Иконка для App Store (обязательно)	512 x 512 Рекомендовано: 1024 x 1024 (высокое разрешение)	512 x 512 Рекомендовано: 1024 x 1024 (высокое разрешение)	“Application Icons” (страница 166)

Описание	Размер для iPhone и iPod touch (в пикселях)	Размер для iPad (в пикселях)	Руководство
Загрузочное изображение (обязательно)	320x480 640x960 (высокое разрешение)	портретная ориентация: 768x1004 1536x2008 (высокое разрешение) горизонтальная ориентация: 1024x748 2048x1496 (высокое разрешение)	“Launch Images” (страница 168)
Маленькая иконка для результатов поиска в Spotlight и Настройках (рекомендовано)	29 x 29 58 x 58 (высокое разрешение)	50 x 50 for Spotlight search results 29 x 29 for Settings	“Small Icons” (страница 171)
Иконка документа (рекомендовано для индивидуальных типов документов)	22 x 29 44 x 58 (высокое разрешение)	64 x 64 320 x 320	“Document Icons” (страница 171)
Иконка веб-клипа (рекомендовано для веб-приложений и веб-сайтов)	57 x 57 114 x 114 (высокое разрешение)	72 x 72	“Web Clip Icons” (страница 175)
Иконки на панели Инструментов и панели Навигации (опционально)	Примерно 20 x 20 Примерно 40 x 40 (высокое разрешение)	Примерно 20 x 20	“Icons for Navigation Bars, Toolbars, and Tab Bars” (страница 176)
Иконки на панели Ярлыков (опционально)	Примерно 30 x 30 Примерно 60 x 60 (высокое разрешение)	Примерно 30 x 30	“Icons for Navigation Bars, Toolbars, and Tab Bars” (страница 176)
Иконка Newsstand cover приложения для App Store по умолчанию (необходимо для Newsstand-приложения)	По крайней мере, 512 пикселей по самому длинному краю. По крайней мере, 1024 пикселей по самому длинному краю (высокое разрешение)	По крайней мере, 512 пикселей по самому длинному краю. По крайней мере, 1024 пикселей по самому длинному краю (высокое разрешение)	“Newsstand Icons” (страница 178)

Примечание: Для всех изображений и иконок рекомендуется PNG-формат.

Стандартный бит глубины для иконок и изображений - 24 бита (8 бит отдельно для красного, зеленого и голубого), плюс 8-и битовый альфа канал.

У вас нет необходимости подстраивать вашу палитру под web-цвета. Хотя вы можете использовать альфа-прозрачность в иконках, которые вы создаете для панели навигации (navigation bar), панели инструментов (toolbar), панели ярлыков (tab bar), вы не можете использовать ее в иконке приложения.

Tips for Designing Great Icons and Images (Советы по Проектированию Великолепной Иконки и Изображения)

Красивые иконки и изображения являются фундаментальной частью iOS пользовательского впечатления (user experience). Не только декоративные иконки и изображения в вашем приложении играют важную роль во взаимодействии с пользователем.

Для достижения наилучших результатов, воспользуйтесь помощью профессионального дизайнера. Опытный дизайнер поможет разработать вам общий визуальный стиль для вашего приложения и применить его на все иконки и изображения в нем.

Используйте универсальные образы, чтобы люди легко их узнавали. Избегайте фокусирования на второстепенных или неясных аспектах элементов.

Выбирайте простое. В частности не используйте множество различных изображений в иконке. Старайтесь использовать один объект, который выражает сущность вашего приложения. Начните с базовой формы и осторожно добавьте детали. Если содержание иконки или формы чрезмерно сложное, детали могут привести к путанице и могут выглядеть грязно при меньших размерах.

Используйте цвет и тень разумно, чтобы помочь иконке рассказать свою историю. Не добавляйте насыщенность, чтобы сделать иконку просто более красочной. Кроме того, гладкие градиенты, как правило, работают лучше, чем резкое разграничение цветов.

В общем, старайтесь не использовать “греческий” текст и волнистые линии в преподнесении текста. Если вы хотите использовать текст в иконке, но не хотите заострять внимание на словах, начните с того, чтобы сделать текст нечитабельным, уменьшите его или удвойте слои.

В общем, создавайте идеализированную версию предмета, вместо использования фотографий. Хотя использование фотографии (или скриншота) в иконке может быть целесообразным, часто лучше усилить реальность художественным способом. Создавайте расширенные версии, которые помогут подчеркнуть аспекты этого вопроса, на которые вы хотите обратить внимание пользователей.

Если ваше приложение имеет узнаваемый интерфейс, рассмотрите вопрос о создании изысканного представления о нем, вместо фактического скриншота вашего

программного обеспечения на иконке приложения. Создавать расширенную версию интерфейса особенно важно, когда пользователи могут перепутать большую версию иконки с фактическим интерфейсом приложения.

Не используйте элементы интерфейса iOS в ваших иллюстрациях. Вы же не хотите, чтобы пользователи спутали ваши иконки или изображения с пользовательским интерфейсом iOS.

Не используйте продукты и товары Apple в иллюстрациях. Символы, которые представляют собой продукты Apple, защищены авторским правом и не могут быть воспроизведены в иконках или изображениях. В общем, это отличная идея, чтобы избежать репродукций любого конкретного устройства в ваших работах, поскольку их дизайн часто меняется и иконки или изображения, основанные на нем, могут отслеживаться датой.

Не используйте иконки iOS-приложения в своем интерфейсе. Это может ввести в заблуждение пользователей, если иконка приложения будет использоваться для обозначения различных вещей, в различных местах по всей системе.

Точно изображайте реальные вещества. Иконки, которые представляют реальные объекты, должны выглядеть так, будто она сделаны из реальных материалов и имеют реальные массы. Реалистичные иконки точно повторяют характеристики веществ, таких как ткань, стекло, бумага, металл, и передают вес и чувства объекта.

Используйте прозрачность, когда это имеет смысл. Прозрачность изображения может помочь отобразить стекло или пластик, но это может быть неубедительным. Вы не должны использовать прозрачность в вашей иконке приложения.

Tips for Creating Great Artwork for the Retina Display (Советы по Созданию Великолепной Художественной Работы для Retina Дисплея)

Retina дисплей позволяет отображать вам в высоком разрешении ваши работы и иконки. Если вы просто расширяете вашу художественную работу, вы упустите возможность обеспечить красивые, увлекательные изображения, ожидаемые пользователями. Вместо этого вы должны переделать существующие изображения для создания больших, лучшего качества версий, в которых:

- **Обогащайте текстуру.** Например, в версии высокого разрешения иконок Settings и Contacts металлические и бумажные текстуры четко видны.



- **Больше деталей.** Например, в версии высокого разрешения иконок Safari и Notes вы можете увидеть детали, такие как точные очертания континентов за компасом и край оторванной бумаги предыдущей записи.

● **Больше реалистичности.** Например, в версии высокого разрешения иконок Compass и Photos сочетаются богатые текстуры и мелкие детали для создания реалистичного образа компаса и фотографии.



Хотя иконки панели проще, чем иконки приложения, вы должны рассматривать вопрос о добавлении деталей, поскольку вы создаете их версии в высоком разрешении. Например, иконка панели ярлыков Artist в Music-приложении представляет обтекаемый силуэт исполнителя. В версии высокого разрешения эта иконка является также узнаваемой, но включает в себя больше деталей.



Следующие методы помогут вам получить отличные результаты при создании художественной работы в высоком разрешении.

Увеличивайте ваше оригинальное изображение на 200% с помощью алгоритма масштабирования “nearest neighbor”. Это отлично работает, если оригинальное изображение не было создано в векторной форме и не включает слой эффектов. Результатом является большое, пикселизированное изображение, поверх которого можно рисовать работу в высоком разрешении. Это хороший способ начать, потому что позволяет сохранить оригинальный макет дизайна. Если оригинальное изображение было создано в векторной форме или оно включает в себя слои эффектов, вы можете использовать стандартный алгоритм масштабирования вместо алгоритма “nearest neighbor”.

Добавляйте детализацию в глубину. Не стесняйтесь делать очень маленькие элементы, потому что в высоком разрешении гораздо больше места для мелких деталей. Например однопиксельная точка исходного изображения становится четырехпиксельной точкой (т.е. 2x2 пикселя) в увеличенной версии.

Рассматривайте смягчение расширенных элементов. Если, например, у вас есть четкая, однопиксельная разделительная линия в оригинальном изображении, то при увеличении изображения она станет двухпиксельной. Но для некоторых линий и элементов вы можете смягчить результат масштабирования или даже оставить элемент меньшего размера.

Пробуйте добавлять размытие для лучших результатов в таких эффектах, как гравюры и тени. Например, текст гравировки обычно создается путем переноса копии изображения текста на 1 пиксель. Масштабирование этого сдвига приведет к гравюре шириной 2 пикселя, которая, вероятно, будет выглядеть очень резкой и нереальной в высоком разрешении. Для улучшения этого, вы можете не изменять сдвиг (т.е. на 1 пиксель), но добавить 1 пиксель размытия, чтобы смягчить гравировку. Это по-прежнему двухпиксельная гравировка, но внешний пиксель теперь больше похож на половину пикселя в ширину, в результате чего сохраняется размерность.

Tips for Creating Resizable Images

(Советы по Созданию Масштабируемого Изображения)

Вы можете создать изменяемый размер изображения, чтобы индивидуализировать фон из нескольких стандартных элементов пользовательского интерфейса, такие как роверы, кнопки, навигационные панели (navigation bars), панели вкладок (tab bars) и панели инструментов (toolbars) (в том числе пункты этих панелей). Предоставление масштабируемых изображений для этих элементов может привести к лучшей производительности приложения.

Для многих элементов пользовательского интерфейса, вы можете также указать заглушки (end caps) в дополнение к внешнему виду фона. Заглушки определяют область изображения, которые не должны масштабироваться. Например, можно создать масштабируемый размер изображения, которое включает в себя четыре заглушки, которые определяют четыре угла кнопки. Когда изображение масштабируется для заполнения области фона кнопки, части, определенные заглушкой, растягиваются без изменений.

В зависимости от размеров масштабируемого изображения, которые вы применяете, iOS либо растягивает, либо заполняет участки области фона элементами пользовательского интерфейса. Под растягиванием подразумевается то, что при масштабировании изображения не учитываются исходные пропорции. Растягивание эффективно, но не желательно для “multipixel” изображений, которые могут искажаться. Заполнение изображения – это многократное повторение оригинального изображения для заполнения целевой области. Заполнение является менее производительным, чем растяжение, но это единственный путь для достижения текстурированного или узорного эффекта.

Как правило, вы должны предоставить наименьшее изображение (за исключением заглушки), что в результате будет выглядеть, как вы этого желаете. Например:

- Используя сплошной цвет в изображении без градиента, вам потребуется изображение 1x1 пиксель.
- Используя вертикальный градиент, создайте изображение шириной 1 пиксель и высотой, которая будет такой же, как и высота фона элемента пользовательского интерфейса.
- Предоставляя повторяющийся текстурированный вид, необходимо создать изображение с размерами, которые соответствуют размерам повторяемой части текстуры.
- Предоставляя неповторяющиеся текстуры, необходимо создать статическое изображение, с размерами, соответствующими размерам области фона элемента пользовательского интерфейса.

Примечание: Если вы создаете масштабируемое изображение для Retina display, вам также необходимо обеспечить версию изображения для высокого изображения. Например, вы могли бы также предоставить однотонные изображения 2x2 пикселя или изображения градиента, имеющие ширину 2 пикселя.

Application Icons (Иконки Приложения)

Иконка приложения - это иконка, которую пользователи располагают на экране устройства и на которую нажимают для запуска приложения. Это место, где брендинг и

четкий визуальный дизайн сходятся вместе в компактной, ясно осознаваемой, привлекательной упаковке. Каждое приложение должно иметь свою иконку.

Примечание: Если ваше приложение является игрой, эта иконка также используется в Game Center.

Попробуйте сбалансировать в вашей иконке привлекательность для глаза и четкость понимания таким образом, чтобы она красиво и четко передавала квинтэссенцию того, что предлагает ваше приложение. Также, хорошо выяснить, как ваш выбор изображения и цвета может быть интерпретирован людьми из других культур.

Создавайте различные размеры вашей иконки приложения для различных устройств. Если вы создаете универсальное приложение, вам необходимо предоставить иконку приложения во всех трех размерах.

Для iPhone и iPod touch необходимы оба этих размера:

- 57x57 пикселей;
- 114x114 пикселей (высокое разрешение).

Для iPad необходим такой размер:

- 72x72 пикселя.

Когда iOS отображает иконку вашего приложения на экране устройства, оно автоматически добавляет следующие визуальные эффекты:

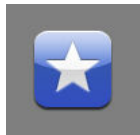
- Закругленные углы;
- Падающая тень;
- Блик (если только вы не убрали этот эффект).

Например, простая иконка приложения под iPhone в размере 57x57 пикселей может выглядеть вот так:



Когда она отображается на экране iPhone, iOS добавляет закругленные углы, падающую тень и блик.

И, таким образом, иконка приложения может выглядеть вот так:



Примечание: Вы можете убрать добавление блика на иконку вашего приложения. Чтобы это сделать, вы должны добавить ключ `UIPrerenderedIcon` в Info.plist файле вашего приложения (чтобы узнать больше об этом файле, смотрите “The Information Property List” в iOS Application Programming Guide).

Присутствие (или отсутствие) добавленного блика не изменяет размеров иконки вашего приложения.

Убедитесь, что ваша иконка соответствует визуальным характеристикам, предоставленным iOS. Вы должны создать изображение, которое:

- Имеет углы в 90°;
- Не имеет какого-либо блеска или свечения (shine or gloss) (если только вы не выбрали удалили добавления блика);
- Не использует альфа-прозрачность.

Задайте вашей иконке приложения заметный фон. Иконки с заметным фоном выглядят изначально лучше на экране iPhone из-за закругленных углов, которые добавляет iOS. Потому что общие для всех закругленные углы делают возможным, чтобы все иконки на экране имели одинаковый внешний вид, и на них было легко нажимать. Если вы создаете иконку с фоном, который исчезнет, когда иконка будет видна на экране, пользователи не увидят закругленных углов. Такие иконки очень часто не выглядят так, что на них можно нажать и имеют тенденцию к нарушению порядка и симметрии на экране устройства, на которую рассчитывают пользователи.

Убедитесь, что ваше изображение полностью заполняет необходимую область. Если границы вашего изображения меньше рекомендованного размера, или вы используете прозрачность для создания областей, с эффектом "просмотра насквозь" ("see-through"), ваша иконка может быть помещена поверх черного фона с закругленными углами.

Например, приложение может предоставить иконку с прозрачным фоном, как на картинке с голубой звездой с левого края. Когда iOS будет отображать эту иконку на экране устройства, она будет выглядеть, как изображение в середине (если не было добавлено блика) или как изображение справа (если был добавлен блик).



Иконка, которая появляется поверх визуально-черного фона, выглядит особенно непривлекательно на экране iPhone, который отображает индивидуализированную фоновую картинку.

Создайте 512x512 пиксельную версию иконки вашего приложения для отображения на App Store. Хотя это и важно, чтобы эта версия четко соответствовала иконке вашего приложения, она может быть более детализированной. Никакие визуальные эффекты не добавляются к этой версии иконки вашего приложения.

Если вы разрабатываете приложение для ad-hoc распространения (то есть только для вас, и не через App Store), вы также должны предоставить 512x512 пиксельную версию иконки вашего приложения. Эта иконка идентифицирует ваше приложение в iTunes.

iOS может также использовать это большое изображение для других нужд. В приложении для iPad, например, iOS использует 512x512 пиксельное изображение для генерации большой иконки документа, если индивидуализированная иконка документа не предоставлена.

Launch Images (Картинки Запуска)

Чтобы улучшить пользовательское впечатление при запуске приложения, вы должны предоставить как минимум одно изображение для картинки запуска. Картинка запуска выглядит очень похоже на первый экран, который отображает ваше приложение. iOS тот час же отображает изображение, когда пользователь запускает приложение, и оно продолжает отображаться до тех пор, пока приложение не будет полностью готово к использованию. Как только ваше приложение готово к использованию, ваше приложение отображает первый экран, который становится на место “изображения-заглушки”.

Предоставьте картинку запуска, чтобы улучшить пользовательский опыт.

Старайтесь не использовать картинку запуска в качестве:

- "Входа в приложение" (“application entry experience”), например, как splashs-screen
- Окна "О приложении"
- Элемента бренда, если он не является статической частью первого экрана вашего приложения

Из-за того, что пользователи обычно быстро переключаются между приложениями, вы должны приложить все усилия, чтобы свести к минимуму время запуска, и вы должны создать дизайн картинки запуска, которое преуменьшает впечатление, чем привлекает внимание.

Обычно, дизайн картинки запуска идентичен с первым экраном приложения.

Исключения:

Текст. Картинка соединения - статична, и таким образом, любой текст, который вы отображаете, на ней не будет локализован.

Элементы UI, которые могут быть изменены. Избегайте вставки элементов, которые могут выглядеть по-другому, когда приложение закончит соединение, ведь пользователи не ожидают увидеть вспышку между картинкой соединения и первым экраном приложения.

Для iPhone и iPod touch картинки запуска включают в себя панель статуса.

Создайте картинки запуска в этих размерах:

- 320x480 пикселей
- 640x960 пикселей (высокое разрешение)

Для iPad картинки запуска не включают в себя панель статуса. Создайте картинки запуска в этих размерах:

Вертикальная ориентация:

- 768x1004 пикселя
- 1536x748 пикселя (высокое разрешение)

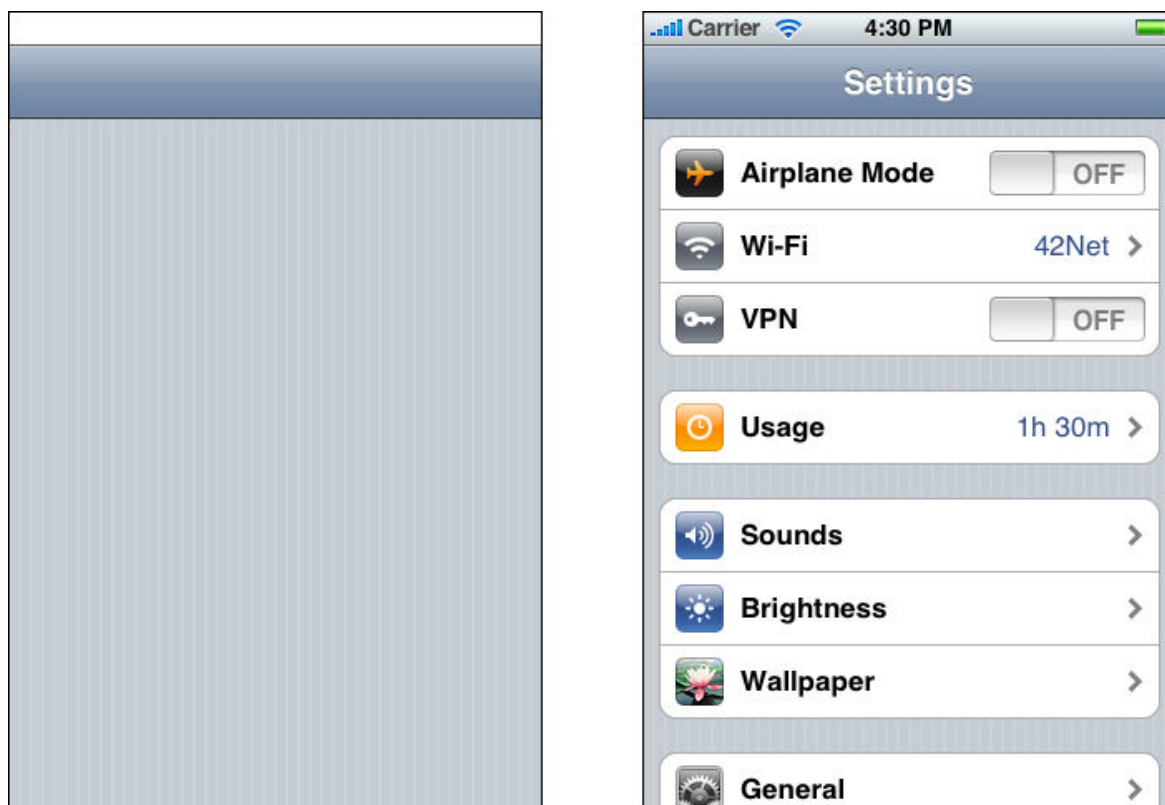
Горизонтальная ориентация:

- 1024x748 пикселя;
- 2048x1496 пикселя (высокое разрешение).

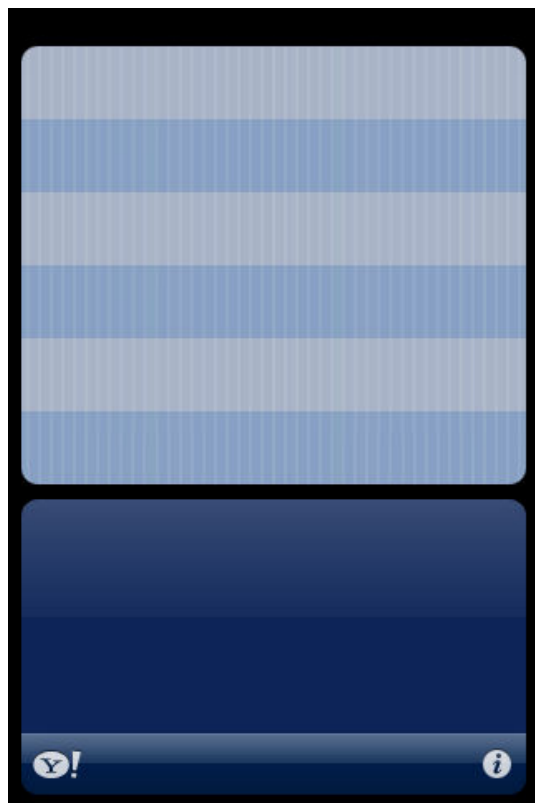
Примечание: большинство iPad-приложений должны предоставлять картинку запуска для каждой ориентации.

Если вы думаете, что следование данному руководству в результате приведет к простой, скучной картинке запуска, то вы правы. Помните, что картинка соединения не означает предоставления дополнительной возможности художественного выражения; она предназначена только для увеличения пользовательского восприятия вашего приложения путем быстрого запуска и немедленной готовности к использованию. Следующие примеры покажут вам насколько простой, может быть, картинка запуска.

Картинка запуска Настроек в iPhone (показана следом за первым экраном приложения) отображает только фон приложения, потому что весь остальной контент приложения гарантированно статичен.



На картинке запуска приложения Stocks для iPhone (показана следом за первым экраном приложения), включены только статические изображения, потому что они всегда видны в первом окне приложения.



Small Icons (Маленькие Иконки)

Каждое приложение должно предоставлять маленькую иконку, которую iOS может отображать, когда имя приложения совпадает со значением в Spotlight (система поиска в продукции Apple). Приложение, которое предоставляет настройки, должно также предоставлять данную иконку для идентификации его во встроенном приложении Настройки (Settings).

Эта иконка должна четко идентифицировать ваше приложение, чтобы пользователи могли опознать его в списке результата поиска или в Настройках.

Для iPhone и iPod touch iOS использует эту же иконку как для результатов поиска в Spotlight, так и в Настройках. Если вы не предоставите эту иконку, iOS может уменьшить вашу иконку приложения для отображения в результатах поиска и в Настройках. Для вашего iPhone приложения создайте две маленьких иконки в размере:

- 29x29 пикселей;
- 58x58 пикселей (высокое разрешение).

Для iPad вы предоставляете отдельные иконки для Настроек и результатов поиска в Spotlight. Создайте две иконки в размере:

- 29x29 пикселей (для Настроек);
- 50x50 пикселей (для результатов поиска в Spotlight).

Примечание: финальный визуальный размер иконки - 48x48 пикселей. iOS обрезает 1 пиксель с каждой стороны вашей работы и добавляет падающую тень (drop shadow). Убедитесь, что вы приняли это во внимание при дизайне вашей иконки для результатов поиска в Spotlight.

Document Icons (Иконки Документа)

Если ваше iOS-приложение создает документы индивидуального типа, вы, возможно, захотите создать индивидуальную иконку, которая идентифицирует этот тип для пользователей. Если вы не предоставляете индивидуальную иконку документа, iOS создает ее по умолчанию, используя иконку вашего приложения (включая добавленные визуальные эффекты).

Например, иконка документа по умолчанию, которая использует 57x57 пиксельную белую звезду, иконка приложения в iPhone будет выглядеть вот так:



Иконка документа в высоком разрешении по умолчанию, которая использует 114x114 пиксельную белую звезду - иконка приложения на iPhone будет выглядеть так:



На большем iPad иконка документа по умолчанию, которая использует 72x72 пиксельную белую звезду, иконка приложения будет выглядеть вот так:



Опционально вы можете предоставить художественную работу для iOS, для использования вместо иконки вашего приложения. Из-за того, что пользователи будут видеть вашу иконку документа в различных местах, лучше всего создать дизайн изображения, которое будет запоминающимся и четко ассоциироваться с вашим приложением. Ваша работа должна быть привлекательной, выразительной и детализированной.

В зависимости от того, работает ли ваше приложение на iPhone или на iPad, вы используете различную спецификацию по созданию данной иконки.

Document Icon Specifications for iPhone (Спецификация Иконки Документа под iPhone).

Для вашего iPhone приложения, создайте иконку документа в двух размерах:

- 22x29 пикселей;
- 22x58 пикселей (высокое разрешение).

Расположите вашу художественную работу внутри каждой прямоугольной области, как

это необходимо. Работа может быть расположена по центру, может быть смещена или может заполнять все пространство.

Например, если вы предоставляете иконку в размере 22x29 пикселей, которая выглядит как картинка слева, iOS создает иконку документа, у которой есть падающая тень, и, таким образом она начинает выглядеть как картинка справа:



Точно также, если вы предоставите 44x58 пиксельную иконку, которая выглядит как изображение слева, iOS создаст иконку документа, у которой будет падающая тень, то есть она будет выглядеть как изображение справа:



Document Icon Specifications for iPad (Требования к Иконке Документа для iPad)

iOS использует два размера иконки документа для iPad приложений: 64x64 пикселя и 320x320 пикселя. Хорошей идеей является создание обоих размеров, таким образом, ваша иконка документа будет выглядеть хорошо в различных контекстах.

Для обоих размеров общие размеры сторон включают в себя определенное количество "подкладки" (padding), оставляющей небольшую "безопасную зону" для вашей художественной работы. Необходимо убедиться, что ваша работа хорошо подходит к этим безопасным зонам, иначе он может быть обрезан или уменьшен.

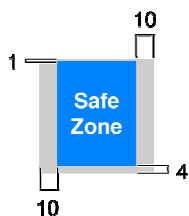
Хотя ваша работа может заполнить всю безопасную зону, правый верхний угол будет всегда частично затемнен эффектом завернутой странички, который добавляет iOS. В дополнение iOS добавляет градиент, который переходит из черного (сверху, чуть ниже эффекта перевернутой странички) в прозрачный (у нижнего края).

Важно: Убедитесь, что вы следуете этой части руководства при создании индивидуализированной иконки документа для вашего iPad-приложения. Если ваша иконка слишком большая или слишком маленькая - в результате иконка документа не будет выглядеть хорошо.

Чтобы создать 64x64 пиксельную иконку документа:

1. Создайте 64x64 пиксельное изображение в формате PNG.
2. Добавьте следующие поля для создания примерного размера безопасной зоны:
 - 1 пиксель сверху;
 - 4 пикселя внизу;
 - 10 пикселей с каждой из сторон.

Ваша безопасная зона должна выглядеть точно так же, как и цветная область показанная ниже:



3. Расположите вашу художественную работу внутри 44x59 пиксельной зоны. Работа может быть выровнена по центру, смещена или может заполнить всю безопасную зону. (Помните, что iOS добавляет эффект завернутой странички в правый верхний угол и градиент, который проходит от завернутой странички к нижнему краю.)

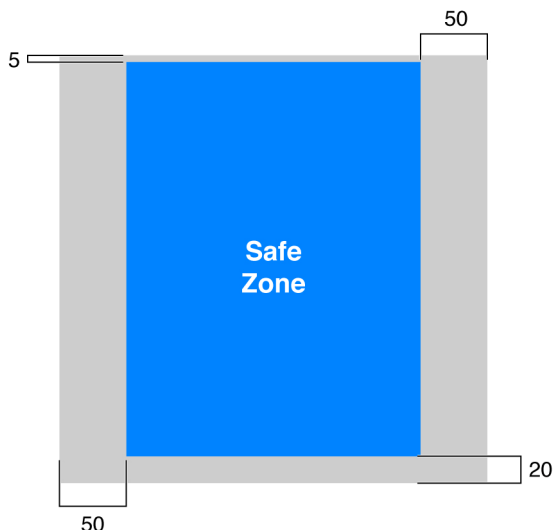
Например, если вы предоставляете иконку, которая выглядит как изображение слева, iOS создаст иконку документа, которая будет выглядеть как изображение справа.



Чтобы создать 320x320 пиксельную иконку документа:

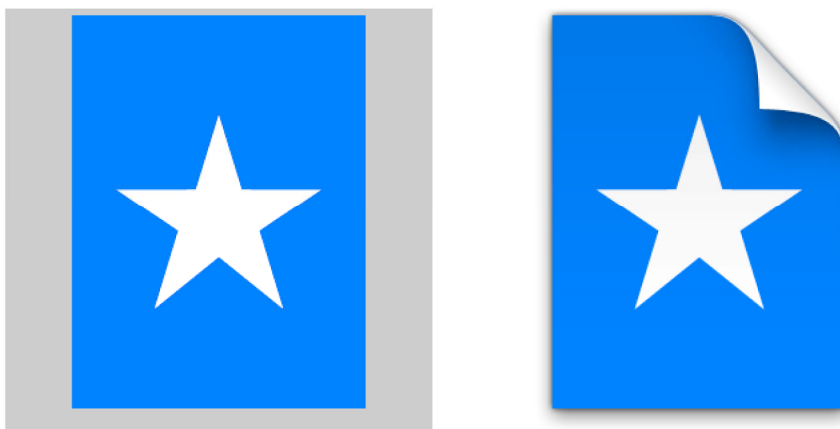
1. Создайте 320x320 пиксельное изображение в формате PNG.
2. Добавьте следующие области, чтобы создать примерный размер защищенной зоны:
 - 5 пикселей сверху;
 - 20 пикселей снизу;
 - 50 пикселей с каждой стороны.

Ваша безопасная зона должна выглядеть, так же как и область, показанная снизу:



3. Расположите вашу работу внутри безопасной зоны, которая в размере 220x295 пикселей. Работа может быть выровнена по центру, смещена или может заполнять всю безопасную зону. (Помните, что iOS добавляет эффект завернутой странички в правый верхний угол и градиент, который проходит от завернутой странички к нижнему краю.)

Например, если вы предоставляете иконку, которая выглядит как изображение слева, iOS создаст иконку документа, которая будет выглядеть как изображение справа.



Web Clip Icons (Web Clip Иконки)

Если у вас есть web-приложение или web-сайт, вы можете предоставить индивидуальную иконку, которую пользователи могут отображать на экране iPhone, используя функцию web clip. Пользователи нажимают на иконку, чтобы перейти к web-контенту быстро. Вы можете создать иконку, которая представляет ваш web-сайт в целом или иконку, которая представляет отдельную web-страницу.

Если ваш web-контент отличается легко узнаваемым изображением или отличительной цветовой схемой, то имеет смысл внедрить его в вашу иконку. Однако, чтобы убедиться, что ваша иконка хорошо выглядит на устройстве, вы также должны следовать руководству в этом разделе. (Чтобы узнать, как добавить код в ваш web-контент, чтобы предоставить индивидуальную иконку, смотрите Safari Web Content Guide.)

Для iPhone и iPod touch создайте иконки в таком размере:

- 57x57 пикселей;
- 114x114 пикселей (высокое разрешение).

Для iPad создайте иконку с такими размерами:

- 72x72 пикселя;
- 114x114 пикселей (высокое разрешение).

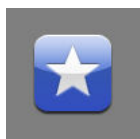
Так же как и с иконками приложения, iOS автоматически добавляет некоторые визуальные эффекты к вашей иконке, которые координируют вашу иконку с теми, которые являются встроенными на устройстве. Более точно, iOS добавляет:

- Закругленные углы;
- Падающую тень;
- Блик.

Например, простая 57x57 пиксельная иконка web-страницы может выглядеть вот так:



Когда ваша 57x57 пиксельная иконка отображается на экране iPhone, iOS придает вот такой вид этой иконке:



Примечание: Вы можете предотвратить добавление всех эффектов, назвав вашу иконку `apple-touch-icon-precomposed.png` (это доступно в iOS 2 и позже).

Убедитесь, что ваша иконка подходит под визуальные дополнения, добавляющиеся iOS (если вы их хотите). Вы должны создать изображение в формате PNG, которое:

- Имеет углы в 90°
- Не имеет бликов.

Icons for Navigation bars, Toolbars, and Tab bars (Иконки для Панелей Навигации, Инструментов и Ярлыков)

Там, где это возможно, вы должны использовать кнопки и иконки, предоставленные системой, чтобы отобразить стандартные задачи в вашем приложении. Для полного списка стандартных иконок и кнопок, а также руководства по их использованию, смотрите “System-Provided Buttons and Icons” (стр 135).

Конечно же, не каждая задача вашего приложения - стандартная. Если ваше приложение поддерживает индивидуальные задачи, которые пользователи должны часто решать, вам необходимо создать индивидуальные иконки, которые представляют эти задачи в вашей панели Инструментов или Навигационной панели. Точно также, если ваше приложение отображает Панель Ярлыков, которая позволяет пользователям переключаться по индивидуальным режимам приложения или индивидуальным множествам данных, вам необходимо сделать дизайн иконок Панели Ярлыков, которые представляют данные режимы или данные.

Прежде чем создавать художественную работу для вашей иконки, вам необходимо потратить немного времени и подумать о том, что она должна выражать. Если вы создаете дизайн, нацельтесь на то, что иконка должна быть:

- **Простой и прямой в понимании.** Слишком много деталей могут сделать иконку слишком неряшливой или неразборчивой.
 - **Не должна быть похожа на ни на одну из иконок, предоставленной системой.** Пользователи должны иметь возможность легко различать вашу индивидуализированную иконку и стандартные иконки.
 - **Должна быть легка в понимании и применении.** Старайтесь создать символ, который большинство пользователей правильно интерпретируют и никому из пользователей она не покажется оскорбительной.
-

Важно: Убедитесь, что вы избежали использования изображений, которые в вашем дизайне копируют продукцию Apple. Эти символы защищены авторским правом и дизайн продукции может часто меняться.

Как только вы определились с внешним видом вашей иконки, следуйте следующим правилам в ее создании:

- Используйте чисто белый цвет с необходимой альфа-прозрачностью.
- Не включайте эффект drop shadow (падающая тень).
- Используйте сглаживание.
- Если вы решили использовать эффект “bevel”, убедитесь, что он в 90° (чтобы вам было легче, представьте, что источник света расположен на верхней части иконки).

Для иконок панелей Инструментов и Навигации, создавайте иконки в следующих размерах:

- Для iPhone, iPod touch и iPad :
 - Около 20x20 пикселей;
 - Около 40x40 пикселей (высокое разрешение).

Для иконок панели Ярлыков, создавайте иконки в следующих размерах:

- Для iPhone, iPod touch и iPad:
 - Около 30x30 пикселей;
 - Около 60x60 пикселей (высокое разрешение).

Примечание: Иконка, которую вы предоставляете для панели Инструментов, Навигации и Ярлыков, используется как маска для создания иконки, которую вы видите в вашем приложении. Нет необходимости создавать полноцветную иконку.

Не включайте отдельно нажатое или выделенное состояние ваших иконок. iOS автоматически предоставляет эти состояния для пунктов в на панели Навигации, Инструментов и Ярлыков - то есть у вас нет необходимости их предоставлять. Из-за того, что эти визуальные эффекты являются автоматическими, вы не можете изменять их внешний вид.

Придайте всем иконкам на панели похожее визуальное восприятие. Нацельтесь на баланс общего размера, уровня детализации и использования четких областей по всем иконкам, которые могут появиться на определенной панели. В общем, очень плохо выглядит комбинирование в одной панели иконок, которые большого размера, квадратные и полностью заполненные, с маленькими детализированными и не заполненными иконками.

Newsstand Icons (Иконки Газетного Киоска)

Если ваше приложение использует Newsstand Kit для публикации периодического контента на основе подписки, вы должны предоставить иконки для отображения на AppStore и на устройствах пользователей. Следуйте инструкциям в этом разделе, чтобы ваши иконки хорошо выглядели на iOS-устройствах во всех контекстах.

Важно: Соотношение сторон всех иконок газетного киоска должны быть между 1:2 и 2:1. Кроме того, все иконки газетного киоска должны быть ровными и иметь углы 90°. Не добавляйте перспективы в иконки газетного киоска.

Все приложения газетного киоска должны предоставлять обложку иконки газетного киоска, которая представляется в виде стандартного дизайна обложки в AppStore. Длинная сторона должна быть 512 пикселей (1024 пикселей для высокого разрешения). Обратите внимание, что эта иконка находится отдельно от иконки приложения, которую все iOS-приложения должны предоставить.

Ваша стандартная иконка обложки газетного киоска должна быть обобщенной факсимильной обложкой типичной версии, которая фокусируется на последовательности от выпуска к выпуску. Например:

- Старайтесь не добавлять к вашей стандартной иконке обложки газетного киоска элементы, которые пользователи никогда не увидят на фактической обложке, например, сообщение “tap here for the latest issue” (“Нажмите здесь для последнего вопроса”).
- Избегайте изображений (artwork) или заголовков, которые являются сезонными или актуальными, таких как изображения, связанные с праздниками или заголовки, которые относятся к текущим событиям.

В частности, не используйте повторно обложку предыдущего номера для используемой стандартной обложки иконки газетного киоска, потому что пользователей может запутать ваше приложение текущей версии.

В дополнение к стандартной иконке обложки газетного киоска, вы должны предоставить отдельную иконку, которая точно отображает каждую новую версию, таким образом, она может появляться на полках газетного киоска и в многозадачной панели (multitasking bar) на iOS-устройствах. В отличие от стандартной иконки обложки, каждая такая иконка должна отображать информацию о содержании конкретной версии.

Эта рекомендация к созданию одной большой иконки для каждой версии, для ее масштабирования iOS, для отображения в обоих местах (иконка отображается на полке газетного киоска в большем размере, чем в многозадачной панели). В зависимости от точного местоположения той или иной версии на полке газетного киоска, iOS также может добавить перспективу иконки, чтобы она соответствовала реальному взгляду на полку.

В частности необходимо создать иконку для каждой версии, размер длинного края которой должен быть по крайней мере 512 пикселей (1024 пикселей для высокого разрешения). Для отображения иконки текущей версии на полке газетного киоска и в многозадачной панели, iOS масштабирует вашу большую иконку в следующие размеры:

Table 8-2 Максимальные размеры длинного края масштабируемой иконки для каждой версии

Устройство	Размер масштабируемого длинного края (полка газетного киоска)	Размер масштабируемого длинного края (многозадачная панель)
iPhone и iPod touch	90 пикселей (180 пикселей для высокого разрешения)	57 пикселей (114 пикселей для высокого разрешения)

Устройство	Размер масштабируемого длинного края (полка газетного киоска)	Размер масштабируемого длинного края (многозадачная панель)
iPad	126 пикселей (252 пикселя для высокого разрешения)	72 пикселя (114 пикселей для высокого разрешения)

В дополнение к предоставлению иконки, используйте ключи в Info.plist файле вашего приложения для определения отображения иконки на устройстве. Во-первых, используйте обязательный тип ключа, чтобы указать, будет ли ваш контент для журнала или для газеты. Во-вторых, используйте привязку ключом для указания визуальных эффектов, которые iOS следует добавить к иконке у нижнего края стандартной газеты. Для получения дополнительной информации об этих ключах и их значении см. раздел “Contents of the UINewsstandIcon Dictionary” в Information Property List Key Reference.

Когда вы указываете тип привязки журнала, iOS добавляет появление нескольких страниц и тень, которая подчеркивает толщину. Вы также должны указать левый или правый край переплета для иконки журнала, который получит обязательный сшитый внешний вид.

Например, предположим, что вы предоставляете следующую иконку:



Если вы укажете тип переплета слева, iOS добавит обязательное появление сшивания к левому краю и несколько страниц появятся справа.



При указании газетного типа привязки iOS добавляет появление дополнительных экземпляров газеты, которые укладываются под текущей версией.

Если ваша газета стандартного размера, вы можете задать тип переплета снизу, чтобы придать иконке появление складки на ее нижнем крае. Если ваша газета таблоидного размера (то есть, примерно в два раза меньше плаката), вы можете указать тип переплета слева или справа или без него, чтобы избежать тени во внешнем виде.

Например, предположим, что ваша текущая версия иконки выглядит примерно так:



Если вы укажете тип переплета снизу, iOS добавит появление тени у нижнего края. (iOS добавит сложенную бумагу независимо от того, какое значение вы укажете для связывания ключевого края).



Для получения дополнительной информации о настройке приложения газетного киоска, см. iTunes Connect Developer Guide.