

*Адресуется всем,
кто заботится об удобстве пользователя*

Содержание

| | |
|--|-----------|
| Предисловие | 14 |
| Благодарности | 15 |
| Об авторе | 16 |
| Введение | 17 |
| Часть I. ОСНОВЫ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ | 19 |
| Глава 1. Пользовательский интерфейс Android и материальный дизайн | 20 |
| Краткая история развития дизайна Android | 20 |
| Материальный дизайн | 23 |
| Основные понятия | 23 |
| Взаимодействия и анимация | 26 |
| Типографика | 27 |
| Единицы измерения и выравнивание | 28 |
| Веб-сайт о дизайне Android | 28 |
| Основные принципы | 29 |
| Делайте что-то одно, но делайте это хорошо | 29 |
| Будьте вежливы | 31 |
| Образы, образы, образы | 32 |
| Простое, но мощное | 33 |
| Единообразие с платформой | 35 |
| Покорите пользователя | 36 |
| Стандартные компоненты | 36 |
| Системные панели | 36 |
| Уведомления | 37 |
| Панель приложения | 38 |
| Вкладки и навигационное меню | 40 |
| Плавающая кнопка | 40 |
| Поддержка разных устройств | 41 |
| Как избежать болезненных ошибок | 42 |
| Кнопка «Меню» | 42 |
| Долгое нажатие | 42 |
| Пиктограммы уведомлений | 43 |
| Стили из других платформ | 43 |
| В заключение | 44 |

| | |
|---|-----------|
| Глава 2. Виджеты – строительные блоки пользовательского интерфейса | 45 |
| Что такое виджет..... | 45 |
| Идентификаторы виджетов | 47 |
| Размеры виджетов..... | 48 |
| Отображение текста..... | 52 |
| TextView..... | 53 |
| EditText..... | 56 |
| Button..... | 56 |
| Отображение изображений..... | 57 |
| Фон | 57 |
| ImageView..... | 58 |
| ImageButton..... | 59 |
| Виджеты для ввода информации пользователем..... | 59 |
| Другие примечательные виджеты..... | 63 |
| Прием и обработка событий..... | 66 |
| OnClickListener | 66 |
| OnLongClickListener..... | 66 |
| OnTouchListener..... | 66 |
| Другие обработчики событий..... | 67 |
| В заключение..... | 67 |
| | |
| Глава 3. Создание пользовательских интерфейсов с применением фрагментов и групп компонентов..... | 68 |
| ViewGroup и типовые реализации..... | 68 |
| FrameLayout..... | 69 |
| LinearLayout..... | 69 |
| RelativeLayout..... | 72 |
| AdapterView..... | 74 |
| ListView..... | 74 |
| GridView..... | 76 |
| Spinner..... | 76 |
| Gallery..... | 77 |
| Adapter..... | 77 |
| Интерфейсы для AdapterView | 78 |
| ViewPager..... | 78 |
| Toolbar..... | 79 |
| Другие примечательные реализации ViewGroup..... | 79 |
| Инкапсуляция логики отображения с использованием фрагментов..... | 81 |
| Жизненный цикл фрагментов..... | 82 |
| Передача данных фрагменту..... | 83 |
| Взаимодействие с Activity..... | 85 |
| Транзакции с фрагментами..... | 87 |

| | |
|-------------------------------------|----|
| Обсуждение проблем..... | 88 |
| Библиотека поддержки..... | 89 |
| Библиотека CardView | 91 |
| Библиотека Design | 91 |
| Библиотека GridLayout | 92 |
| Библиотека Leanback..... | 92 |
| Библиотека MediaRouter | 92 |
| Библиотека Palette | 92 |
| Библиотека RecyclerView | 93 |
| Библиотека Support Annotations..... | 93 |
| В заключение | 94 |

| | |
|--|-----------|
| Глава 4. Добавление графики и ресурсов в приложения | 95 |
| Введение в ресурсы ОС Android..... | 95 |
| Квалификаторы ресурсов | 96 |
| Разрешение..... | 101 |
| Поддерживаемые файлы изображений..... | 102 |
| Растровые изображения | 103 |
| Векторные изображения | 104 |
| Изображения в формате Nine-Patch..... | 104 |
| Графические элементы в XML | 106 |
| Список слоев | 107 |
| Список состояний..... | 108 |
| Список уровней..... | 111 |
| TransitionDrawable | 112 |
| InsetDrawable | 112 |
| ClipDrawable..... | 112 |
| ScaleDrawable..... | 113 |
| ShapeDrawable..... | 113 |
| VectorDrawable..... | 115 |
| AnimatedVectorDrawable | 118 |
| RippleDrawable..... | 119 |
| Другие ресурсы..... | 120 |
| Строки | 121 |
| Массивы..... | 122 |
| Цвета | 123 |
| Размеры..... | 124 |
| Анимации | 124 |
| Числовые идентификаторы | 124 |
| Меню | 125 |
| В заключение..... | 126 |

Часть II. ПОЛНЫЙ ПРОЦЕСС ПРОЕКТИРОВАНИЯ И РАЗРАБОТКИ.....127

Глава 5. Начало создания нового приложения128

| | |
|--|-----|
| Методы проектирования | 128 |
| Типичные методы | 128 |
| Проектирование с ориентацией на пользователя | 129 |
| Определение целей..... | 132 |
| Цели пользователей..... | 132 |
| Группы пользователей..... | 133 |
| Цели продукта..... | 134 |
| Поддержка устройств и конфигураций | 135 |
| Высокоуровневая диаграмма | 136 |
| Эскизы | 139 |
| Начните с навигации | 142 |
| Информационное наполнение | 145 |
| Эскиз страницы с описанием..... | 147 |
| Поддержка нескольких устройств | 149 |
| Соглашения об именовании..... | 150 |
| Заготовки для ресурсов | 151 |
| В заключение | 152 |

Глава 6. Макетирование и разработка основы приложения153

| | |
|--|-----|
| Организация экранов и фрагментов | 153 |
| Создание первого прототипа | 154 |
| Вкладки | 156 |
| Навигационное меню | 158 |
| Представление инструментов | 161 |
| Фрагменты для вкладок | 166 |
| Сведения об инструменте | 180 |
| Оценка первого прототипа | 184 |
| Работа с пользователями | 184 |
| Свободное исследование..... | 185 |
| Специализированные цели | 185 |
| Динамические цели..... | 186 |
| Отзывы пользователей | 186 |
| Следующие шаги..... | 188 |
| В заключение | 189 |

Глава 7. Визуальный дизайн190

| | |
|-----------------------------------|-----|
| Эскизы и графический дизайн | 190 |
| Инструменты..... | 191 |
| Стили..... | 192 |

| | |
|---|------------|
| Скевоморфизм | 192 |
| Минимализм..... | 192 |
| Плоский дизайн..... | 193 |
| Материальный дизайн | 193 |
| Освещение | 194 |
| Цвета..... | 195 |
| Теория | 195 |
| Выбор цветов..... | 198 |
| Приложение выбора столярных инструментов..... | 202 |
| Текст..... | 202 |
| Контраст текста | 203 |
| Размер текста, стили оформления и заглавные буквы..... | 204 |
| Интервалы в тексте | 204 |
| Тени, отбрасываемые текстом | 205 |
| Нестандартные шрифты..... | 205 |
| Доступный словарь | 207 |
| Другие вопросы | 207 |
| Разный объем текста..... | 207 |
| Размеры и доступность изображений..... | 208 |
| Прозрачность и правило 3× | 208 |
| Стандартные пиктограммы | 209 |
| Навигация и визуализация переходов | 209 |
| Обработка ошибочных ситуаций | 210 |
| Пошаговое оформление..... | 210 |
| В заключение..... | 215 |
| Глава 8. Реализация дизайна..... | 216 |
| Совместная работа с дизайнером | 216 |
| Нарезка графических ресурсов | 217 |
| Простые нарезки | 218 |
| Изображения в формате Nine-Patch | 219 |
| Создание альтернативных ресурсов для других размеров..... | 223 |
| Темы и стили..... | 224 |
| Деление дизайн-макета на виджеты..... | 225 |
| Разработка приложения выбора столярных инструментов..... | 227 |
| Главный экран..... | 227 |
| Список инструментов..... | 236 |
| Описание отдельных инструментов..... | 249 |
| Тестирование на разных типах устройств | 253 |
| В заключение | 254 |
| Глава 9. Украшение анимацией..... | 255 |
| Назначение анимации | 255 |
| Анимация виджетов | 256 |

| | |
|---|-----|
| Анимация свойств..... | 257 |
| Управление анимацией свойств..... | 258 |
| Обработчики..... | 258 |
| Типы вычисляемых значений..... | 259 |
| Интерполяция во времени..... | 262 |
| Ключевые кадры | 264 |
| Класс ViewPropertyAnimator..... | 265 |
| Анимация вывода сообщений об ошибках в формах | 265 |
| Анимация пиктограмм..... | 269 |
| Анимация векторных пиктограмм..... | 269 |
| Анимация растровых пиктограмм | 274 |
| Простые переходы | 275 |
| Переходы между сценами..... | 276 |
| Переходы между экранами..... | 279 |
| Переходы в виде расширяющегося круга..... | 283 |
| В заключение..... | 286 |

Часть III. ДОПОЛНИТЕЛЬНЫЕ ВОПРОСЫ СОЗДАНИЯ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ287

| | |
|---|------------|
| Глава 10. Продвинутые приемы | 288 |
| Выявление задержек..... | 288 |
| Использование SysTracer для определения причин задержек | 290 |
| Оптимизация изображений..... | 297 |
| Сжатие изображений..... | 297 |
| Использование изображений подходящих размеров | 299 |
| Кэширование изображений..... | 303 |
| Дополнительные методы увеличения производительности..... | 309 |
| Управление сборкой мусора..... | 309 |
| Шаблон «Держатель виджетов»..... | 310 |
| Устранение избыточного рисования | 312 |
| Обозреватель иерархии | 316 |
| Поиск пропавших виджетов | 318 |
| Устранение ненужных виджетов | 319 |
| Экспорт в формат PSD | 319 |
| Нестандартные шрифты..... | 320 |
| Комплексные виджеты TextView | 323 |
| Имеющиеся промежутки | 323 |
| Использование промежутков для отображения текста со сложным оформлением..... | 325 |
| Класс RecyclerView | 326 |
| Диспетчер размещения..... | 327 |
| Адаптер..... | 327 |

| | |
|-------------------------------|-----|
| Аниматор элементов | 328 |
| Декорирование элементов | 328 |
| В заключение | 328 |

Глава 11. Холсты и дополнительные приемы отображения графики

330

| | |
|--|-----|
| Создание собственных графических элементов | 330 |
| Основная идея | 330 |
| Другие важные методы | 331 |
| Класс Paint | 332 |
| Класс Canvas | 332 |
| Работа с текстом | 333 |
| Простой текстовый элемент | 333 |
| Улучшенный текстовый элемент | 335 |
| Работа с изображениями | 338 |
| Цветовые фильтры | 340 |
| Простой цветовой фильтр | 340 |
| Цветовая матрица | 341 |
| Наложение изображений с помощью режимов PorterDuff | 343 |
| Шейдеры | 353 |
| Закругление краев изображений | 353 |
| Эффект градиентного растворения | 356 |
| В заключение | 359 |

Глава 12. Разработка собственных виджетов

360

| | |
|---|-----|
| Основные идеи | 360 |
| Измерение размеров | 361 |
| Размещение | 361 |
| Рисование | 362 |
| Сохранение и восстановление состояния | 363 |
| Создание собственного виджета | 363 |
| Измерение | 365 |
| Размещение дочерних виджетов | 366 |
| Создание изображения | 366 |
| Сохранение и восстановление состояния | 368 |
| В заключение | 375 |

Глава 13. Обработка ввода и прокрутки

376

| | |
|--|-----|
| Сенсорный ввод | 376 |
| Другие формы ввода | 378 |
| Создание собственного виджета | 378 |
| Создание первоначальных файлов виджета | 379 |
| Измерение | 384 |

| | |
|--|------------|
| Рисование | 387 |
| Подготовка к обработке событий сенсорного ввода..... | 392 |
| Обработка событий сенсорного ввода..... | 396 |
| Прочие вопросы | 412 |
| В заключение | 412 |
| Приложение А. Ресурсы Google Play | 413 |
| Описание приложения..... | 413 |
| Список изменений..... | 414 |
| Пиктограмма приложения..... | 415 |
| Простой способ создания пиктограмм | 416 |
| Создание пиктограмм вручную | 417 |
| Скриншоты..... | 421 |
| Основной баннер..... | 423 |
| Рекламный баннер..... | 425 |
| Видео (YouTube) | 426 |
| Продвижение приложения | 426 |
| Amazon Appstore | 427 |
| Приложение В. Справочник по типичным задачам..... | 428 |
| Скрытие экранной клавиатуры | 428 |
| Включение полноэкранного режима..... | 429 |
| Предотвращение выключения экрана | 430 |
| Определение физических размеров экрана устройства | 430 |
| Определение размеров экрана в пикселях | 431 |
| Определение разрешения экрана | 431 |
| Проверка подключения к сети | 432 |
| Проверка, является ли текущий поток выполнения потоком пользовательского интерфейса | 433 |
| Нестандартные атрибуты виджетов..... | 434 |
| Предметный указатель..... | 441 |

Предисловие

Развитие Android идет с невероятной скоростью, и не отстать от этого развития – сложная задача для любого разработчика. В гонке за последними нововведениями и изменениями в API легко упустить из виду архитектурные изменения в Android. Когда компания Google анонсировала принципы материального дизайна (Material Design), даже проектировщики, долгое время обходившие Android стороной, начали обращать на него внимание.

Разработчики для Android как никто другой должны знать и понимать основные аспекты проектирования интерфейсов и идеи материального дизайна; однако, не имея многолетнего опыта проектирования, может быть сложно разобраться во всем этом. Данная книга проведет вас через процесс проектирования, от абстрактной идеи и набросков на бумаге до воплощения анимационных эффектов, применения RenderScript и создания собственных виджетов. Идея состоит в том, чтобы коснуться каждого базового аспекта и охватить его достаточно полно, чтобы вы могли общаться с дизайнерами интерфейсов на профессиональном уровне или даже создавать интерфейсы самостоятельно.

Дизайнеры решают множество задач, но наиболее важными являются удобство использования и внешняя привлекательность. Вы должны так спроектировать интерфейс своего приложения, чтобы можно было запустить его и начать пользоваться им безо всяких усилий, потому что пользователи мобильных устройств более нетерпеливы, чем пользователи любых других платформ. Пользователи должны точно знать, что смогут взаимодействовать с приложением и будут иметь возможность делать это в спешке. Как следствие вы должны помнить, что соглашения для платформы разрабатывались специально для того, чтобы пользоваться преимуществами усвоенного поведения.

Коль скоро вы взяли эту книгу в руки, я, вероятно, не должен долго объяснять, как важен дизайн пользовательских интерфейсов. Вы и так знаете это. И у вас уже есть желание приступить к созданию удобных и привлекательных приложений.

Эта книга послужит учебным пособием по проектированию и реализации пользовательских интерфейсов, а также удобным справочником, к которому вы будете возвращаться снова и снова. Вы узнаете, как общаться с дизайнерами и разработчиками на их языке, чтобы способствовать улучшению приложений. Вы сможете создавать приложения с визуально привлекательным интерфейсом, легко поддающимся изменениям даже в самые последние минуты перед выпуском.

В конце концов, и разработчики, и дизайнеры желают, чтобы их приложения пользовались популярностью, и я буду рад рассказать вам, как этого добиться.

– Ян Клифтон (Ian G. Clifton)

Благодарности

Может показаться, что второе издание книги дается проще ее автору, но представьте, что вам требуется переписать 90% текста, потому что технологии и направления проектирования ушли далеко вперед, и вы поймете, что без помощи других вам придется очень трудно. Ответственный редактор Лаура Левин (Laura Lewin) еще раз помогла мне не выбиться из графика, даже когда я решил реорганизовать книгу и чересчур углубился в это занятие. Оливия Бесижио (Olivia Basegio), помощник редактора, постоянно следила за изменениями и попутно публиковала фрагменты рукописи, чтобы заинтересованные читатели могли знакомиться с книгой по мере ее развития. Сонглин Кью (Songlin Qiu) снова сыграла роль редактора-консультанта и взяла на себя тяжкий труд привести к удобочитаемому виду главы, написанные ночью. Я также бескрайне признателен моим техническим рецензентам: Адаму Портеру (Adam Porter), Кэмерону Банга (Cameron Banga) и Джошуа Джеймисону (Joshua Jamison), чьи отзывы помогли существенно улучшить эту книгу.

Об авторе

Ян Клифтон (Ian G. Clifton) – профессиональный разработчик приложений для Android, проектировщик пользовательских интерфейсов и автор книг. Работал со многими разработчиками и дизайнерами и руководил подразделением Android, создавшим такие известные приложения, как Saga, CNET News, CBS News и др.

Любовь к технологиям, искусству и своей работе вела Яна разными путями. Помимо приложений для Android, он также занимается разработкой настольных и веб-приложений. В свое время он служил в военно-воздушных силах США специалистом по спутниковым широкополосным системам телеметрии и тогда же создал немало художественных произведений ручкой, углем, кистью и камерой.

Вы можете последовать за Яном в Twitter (<http://twitter.com/ianGClifton>) и познакомиться с его мыслями о разработке мобильных приложений в блоге <http://blog.iangclifton.com>. Он также публикует видеоматериалы в серии «The Essentials of Android Application Development LiveLessons, 2nd Edition», доступные по адресу: <http://goo.gl/4jr2j0>.

Введение

Кому адресована эта книга

Эта книга адресована в первую очередь разработчикам для Android, желающим лучше понять, как создавать пользовательские интерфейсы (ПИ) в Android. Чтобы сосредоточить все внимание на наиболее важных темах проектирования ПИ для Android, в этой книге предполагается, что вы уже имеете элементарные представления об Android. То есть если вы еще не написали своего первого приложения «Hello, World» для Android или не подготовили компьютера для разработки, вы должны сделать это, прежде чем продолжить чтение (в этом вам поможет сайт для Android-разработчиков: <http://developer.android.com/training/basics/firstapp/index.html>).

Большинство разработчиков не имеет или имеет сильно ограниченный опыт дизайна пользовательских интерфейсов, поэтому в этой книге не делается никаких предположений о ваших знаниях в этой области. Всякий раз, когда мы будем сталкиваться с решением, важным с точки зрения дизайна, таким как выбор цвета для того или иного элемента, мы детально опишем, что лежит в основе такого решения, чтобы вы могли осознанно подходить к выработке своих решений.

Организация книги

Эта книга разбита на несколько частей. В первой части «Основы пользовательских интерфейсов» дается обзор пользовательского интерфейса Android и основных направлений проектирования, прежде чем погрузиться в изучение классов, используемых для создания интерфейсов. Она также охватывает приемы использования графики и ресурсов. Во второй части «Полный процесс проектирования и разработки» описываются этапы разработки приложений, от основных идей и целей, через макеты и прототипы, до разработки законченных приложений, включая эффективные схемы размещения элементов, анимационные эффекты и многое другое. В третьей части «Дополнительные вопросы создания пользовательских интерфейсов» исследуются намного более сложные темы, включая диагностику причин низкой производительности пользовательских интерфейсов с помощью Systrace и создание нестандартных виджетов, поддерживающих рисование графических изображений, прокрутку и сохранение информации о состоянии.

В этой книге имеются два приложения. В первом описываются ресурсы Google Play (и охватываются отличия от Amazon Appstore, которые необходимо знать, чтобы подготовить приложение к размещению в обоих электронных магазинах) и приемы создания пиктограмм, представляющих приложения. Второе охватывает общие задачи, связанные с пользовательскими интерфейсами, не укладываемые ни в одну из глав в книге (такие как пользовательские атрибуты виджетов).

На протяжении всей книги основной акцент делается на простоте и ясности реализации. Не нужно бояться сложных тем, таких как создание матриц для трехмерных

преобразований в OpenGL; мы не будем затрагивать их, вместо этого основное внимание уделим созданию гладких анимационных эффектов, добавлению поддержки режимов PorterDuff в виджеты и эффективной обработке событий сенсорного экрана. Мы максимально упростим используемые математические приемы. Кроме того, многочисленные иллюстрации сделают сложные примеры более ясными и практичными.

Как читать эту книгу

Эта книга начинается с обширного обзора перед погружением в специальные и сложные темы. Она задумана так, чтобы читать ее по порядку, но вместе с тем она с успехом может использоваться как справочник. Даже если вы опытный разработчик, читайте главы по порядку, потому что в них охватываются самые разные темы; но при желании вы можете сразу переходить к темам, которые представляют для вас наибольший интерес. Например, если вы действительно хотите сосредоточиться на создании собственных виджетов, вы можете сразу перейти к главе 12 «Разработка собственных виджетов».

Веб-сайт книги

Исходный код примеров из книги можно найти по адресу: <https://github.com/IanG-Clifton/aid2>, а также на сайте издательства: <http://www.informit.com/store/android-user-interface-design-implementing-material-9780134191409>. По этим адресам вы можете создать полную копию репозитория, загрузить ZIP-файл со всеми примерами или просматривать отдельные файлы.

Типографские соглашения в книге

В этой книге используются типографские соглашения, обычные для книг по программированию. Элементы программного кода, такие как имена классов или ключевые слова, оформляются моноширинным шрифтом.

Иногда слишком длинные строки с программным кодом, не уместающиеся по ширине страницы, содержат знак стрелки (➡), показывающий, что строка продолжается ниже.

Иногда вам будут встречаться врезки с полезной информацией, неуместной в основном тексте.



Так оформляются короткие примечания с дополнительной информацией по обсуждаемой теме, которая может вам пригодиться.



Так выглядят советы по обсуждаемой теме.



Потенциальные проблемы, связанные с безопасностью или вероятностью потери данных. Предупреждения, оформленные так, призваны привлечь ваше внимание к потенциальным проблемам, с которыми вы можете столкнуться.

Часть I

ОСНОВЫ ПОЛЬЗОВАТЕЛЬСКИХ ИНТЕРФЕЙСОВ

Пользовательский интерфейс Android и материальный дизайн

Считаю полезным дать обзор пользовательского интерфейса (ПИ) ОС Android, поэтому мы начнем с него. Здесь вы узнаете краткую историю дизайна пользовательского интерфейса в Android и увидите, как развивалась идея материального дизайна, прежде чем оформилась в виде нескольких принципов проектирования. Вы познакомитесь также с некоторыми высокоуровневыми элементами дизайна Android и изменениями, произошедшими в дизайне Android с ростом популярности мобильной операционной системы.

Краткая история развития дизайна Android

Операционная система Android начиналась как чисто технический проект, в рамках которого был проделан удивительно большой объем работы, чтобы превратить ее в платформу, на которой могли бы работать самые разные устройства, имеющие свои технические особенности. Эта основа позволила ОС Android обслуживать многие типы устройств ввода (трекболы, джойстики, выдвижные клавиатуры, сенсорные экраны и т. д.). Она также позволила ОС Android сфокусироваться на масштабируемом дизайне, гораздо более близком напоминающем «резиновую» веб-верстку, чем типичный современный мобильный дизайн. Программный код, составляющий основу системы, мог даже выполняться на устройствах, не имеющих графического процессора (Graphics Processing Unit, GPU). К сожалению, все это означало также, что первоначальный дизайн для Android был чересчур скупым и ориентированным на самые слабые устройства, поэтому разного рода украшения и анимация были редким явлением. Цвета были мягкими и часто плохо согласующимися между собой, а визуальная организация опиралась на разработки прошлых лет и медленно двигалась вперед.

В 2010 году компания Google наняла Матиаса Дуарте (Matias Duarte, более известного своей работой над WebOS) на должность старшего директора по пользо-

вательскому интерфейсу Android, что ясно показало, насколько серьезно относятся в Google к пользовательскому интерфейсу и визуальному дизайну. Бета-версия Android была выпущена задолго до этого, в 2007 году, поэтому перед Матиасом с коллегами был непочатый край работы. Как перейти от очень функционального, но визуально скудного пользовательского интерфейса к интерфейсу, усиливающему функциональность продуманным и качественным дизайном?

Примерно год спустя вышел первый планшетный компьютер на ОС Android версии «Honeycomb» (Android 3.0). Эти планшетные компьютеры дали компании Google настоящую возможность экспериментировать с пользовательским интерфейсом, потому что прежде не существовало версии Android для планшетных компьютеров и, соответственно, у пользователей не было серьезных требований. С радикально новой темой «Holo» эти планшетные компьютеры стали существенным шагом, уводящим от предыдущих стилей оформления Android.

В конце 2011 года компания Google представила версию Android 4.0 «Ice Cream Sandwich», которая показала, насколько удалось усовершенствовать оформление «Honeycomb» для планшетных компьютеров, скрыть некоторый «технический привкус» и улучшить восприятие пользователями. Было устранено деление планшет/телефон, и платформа получилась более единообразной. Изменился даже шрифт по умолчанию – вместо шрифтов Droid стали использоваться шрифты Roboto, имеющие более привлекательный внешний вид. Не важно, являетесь вы поклонником рубленых шрифтов без засечек или нет, вы не сможете не оценить то внимание к деталям (о чем говорит смена шрифта), которое было проявлено.

В 2012 году на конференции Google I/O была представлена обновленная версия Android 4.1 «Jelly Bean». Основной целью этого выпуска было увеличение удобства использования – «Jelly Bean» придала Android намного более привлекательный дизайн. Технология «Project Butter» еще больше улучшила адаптивность пользовательского интерфейса и привнесла в графическую подсистему такие усовершенствования, как тройная буферизация. Обновились даже компоненты Android, к которым разработчики не прикасались годами, такие как всплывающие уведомления. Спустя всего несколько месяцев вышла версия Android 4.2. В ней появился многопользовательский режим, а также поддержка технологий «Daydream» (по сути – хранители экрана, оформленные в виде приложений и обладающие интерактивными функциями), «Photo Sphere» (обеспечивает создание фотопанорам, охватывающих все 360 градусов) и беспроводного зеркалирования. Последовавшая за ней версия Android 4.3 добавила еще больше возможностей, включая поддержку OpenGL ES 3.0. И версия Android 4.4 завершила линейку 4.x, добавив существенные усовершенствования, позволяющие системе Android работать на устройствах с небольшим объемом ОЗУ, до 512 Мбайт.

Затем разработчики внесли радикальные изменения, вылившиеся в создание версии Android 5.0, «Lollipop», в которой внедрили технологию материального дизайна. Посмотрите на рис. 1.1, как менялся дизайн домашнего экрана между версиями Android 1.6 и Android 5.0.

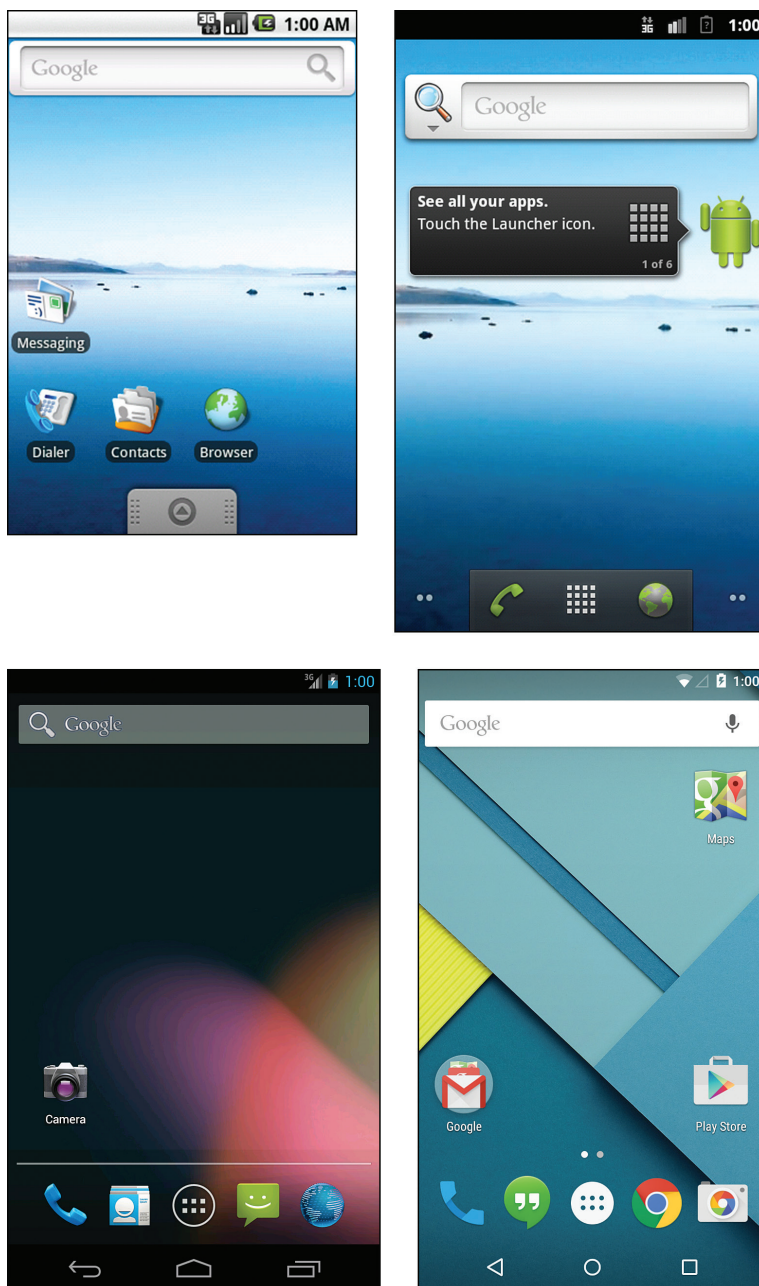


Рис. 1.1 ❖ Домашний экран в версиях Android 1.6 (вверху слева), 2.3 (вверху справа), 4.2 (внизу слева) и 5.0 (внизу справа)

Материальный дизайн

Несмотря на тесную связь между Android 5.0 и материальным дизайном, материальный дизайн (Material Design) – это не только язык визуальных образов для Android 5.0 и выше, но также набор принципов проектирования, применимых к любым типам устройств и произвольным версиям программного обеспечения. То есть принципы материального дизайна должны применяться также к старым версиям Android и даже к веб-приложениям. Google так описывает эту концепцию:

Метафора материального мира – это теория, обобщающая рационализацию пространства и систему движений. Материальный дизайн основывается на ощущениях, возникающих при работе с бумагой и чернилами, но более развит технологически и открыт для воображения и магии.

Если вы рассуждаете как большинство далеких от профессии дизайнера, первой вашей реакцией наверняка будет вопрос: «Что?» Это так просто – отвергнуть подобные заявления, посчитав их обычным «дизайнерским трепом», и не оценить их смысл, но у дизайнеров имеются свои руководящие принципы, цель которых – повысить качество результатов их труда, так же как у обычных разработчиков (вспомните, например, принцип: «Не повторяйся»). В описании выше говорится, что материальный дизайн основан на ощущениях, возникающих при работе с бумагой и чернилами, но он не ограничивается только этими элементами физического мира.

Основные понятия

В мире материального дизайна под бумагой понимается прежде всего поверхность, на которой находится все остальное. Она может растягиваться и сжиматься, в отличие от листа бумаги в реальном мире. То есть лист бумаги может появиться в центре экрана, увеличиваясь из точки до размеров экрана и даже изменяя форму. Лист бумаги всегда появляется с некоторым эффектом (возникает из ничего, постепенно увеличиваясь в размерах или выдвигаясь из-за края экрана) – он никогда не появляется внезапно. Листы бумаги могут надвигаться друг на друга, образуя стопку, делиться пополам, объединяться. Листы бумаги не могут проходить сквозь друг друга, но один лист может скользить поверх другого.

Тот факт, что один лист бумаги может скользить поверх другого, означает, что материальный дизайн существует в трехмерном окружении. Третье измерение, ось Z , определяет, как близко к поверхности экрана находится объект, как накладываются тени, отбрасываемые объектами, и как объекты перекрывают друг друга. Материальный дизайн придает оси Z очень ограниченное значение, в том смысле, что не позволяет отображать сколько-нибудь существенную глубину (то есть не позволяет показать, насколько толстой получилась стопка из листов бумаги на экране). Листы бумаги всегда имеют толщину в один пиксель, не зависящий

от разрешения экрана (идея пикселей, не зависящих от разрешения, – Density-Independent Pixels, или *dip*, – рассматривается в главе 4 «Добавление графики и ресурсов в приложения», а пока рассматривайте *dip* как некую единицу измерения), то есть никогда не изгибаются и не скручиваются.

Устройства не имеют трехмерных экранов (пока), поэтому ощущение глубины создается традиционными приемами перспективного отображения, заслонения и затенения. Как показано на рис. 1.2, объект, находящийся ближе к поверхности, выглядит больше, чем тот же объект, удаленный от экрана. Объект, находящийся выше, заслоняет объекты, находящиеся под ним, как показано на рис. 1.3. Ближний объект отбрасывает тень, как показано на рис. 1.4. Объединение этих простых принципов позволяет получить более четкое ощущение глубины, как показано на рис. 1.5.

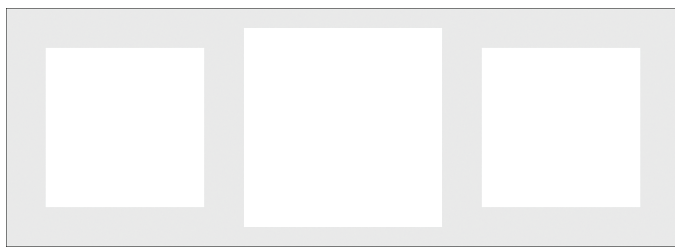


Рис. 1.2 ❖ Под перспективой понимается, что более близкие объекты выглядят больше, но одной перспективы недостаточно, так как не ясно, действительно ли один объект ближе других или он просто больше в размерах

Тени в материальном дизайне создаются светом от двух источников: основным и рассеянным. Представьте, что вы фотографируете кого-то своим телефоном, вспышка даст основной свет, а все остальное освещение даст рассеянный свет. Основной свет создает резкие тени в одном направлении. Рассеянный свет создает размытые тени во всех направлениях. Основной свет исходит из центра сверху над экраном и создает тени, отбрасываемые листами бумаги вниз; кроме того, листы бумаги, находящиеся в нижней части экрана, отбрасывают более длинные тени, потому что свет падает на них под более острым углом. Такого рода детали визуального отображения остаются почти незамеченными, но увеличивают целостность восприятия искусственной среды и реалистичность отбрасываемых теней. Чтобы лучше понять, что это означает, взгляните на рис. 1.6, демонстрирующий описанные идеи на трехмерном изображении.

Конец ознакомительного фрагмента.

Приобрести книгу можно

в интернет-магазине

«Электронный универс»

e-Univers.ru