**Лекция 3:**

# Основы разработки интерфейсов мобильных приложений

Для подготовки данной лекции в качестве основного источника информации была использована книга "Алан Купер об интерфейсе"[[11](http://www.intuit.ru/studies/courses/12643/1191/literature#literature.11)], однако задекларированные в ней принципы были переработаны в контексте программирования для мобильных устройств, а примеры заменены на более подходящие в рамках данного курса. Скриншоты приложений взяты из магазина приложений Google *Play* или сделаны самостоятельно с использованием смартфона Мегафон SP-A20i Mint на платформе Intel Medfield.

Презентацию к данной лекции можно скачать [здесь](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/Lec3_PP.pptx).

**5.1 Визуальный дизайн интерфейсов**

Силы, вложенные в разработку модели поведения программного продукта, будут потрачены впустую, если вы не сумеете должным образом донести до пользователей принципы этого поведения. В случае мобильных продуктов это делается визуальными средствами - путем отображения объектов на дисплее (в некоторых случаях целесообразно использовать тактильные ощущения от нажатия).

Визуальный *дизайн* интерфейсов - очень нужная и уникальная дисциплина, которую следует применять в сочетании с проектированием взаимодействия и промышленным дизайном. Она способна серьезно повлиять на эффективность и привлекательность продукта, но для полной реализации этого потенциала нужно не откладывать визуальный *дизайн* на потом, а сделать его одним из основных инструментов удовлетворения потребностей пользователей и бизнеса.

**5.1.1 Изобразительное искусство, визуальный дизайн интерфейсов и прочие дисциплины дизайна**

Художники и визуальные дизайнеры работают с одними и теми же изобразительными средствами, однако их деятельность служит различным целям. Цель художника - создать объект, взгляд на который вызывает эстетический отклик. Изобразительное искусство - способ самовыражения художника. Художник не связан почти никакими ограничениями. Чем необычнее и своеобразнее продукт его усилий, тем выше он ценится.

Дизайнеры создают объекты, которыми будут пользоваться другие люди. Если говорить о дизайнерах визуальных интерфейсов, то они ищут наилучшее представление, доносящее информацию о поведении программы, в проектировании которой они принимают участие. Придерживаясь целеориентированного подхода, они должны стремиться представлять поведение и информацию в понятном и полезном виде, который поддерживает маркетинговые цели организации и эмоциональные цели персонажей. Разумеется, визуальный дизайн пользовательских интерфейсов не исключает эстетических соображений, но такие соображения не должны выходить за рамки функционального каркаса.

**5.1.2 Графический дизайн и пользовательские интерфейсы**

Графические дизайнеры обычно очень хорошо разбираются в визуальных аспектах и хуже представляют себе понятия, лежащие в основе поведения программного продукта и взаимодействия с ним. Они способны создавать красивую и адекватную внешность интерфейсов, а кроме того - привносить фирменный стиль во внешний вид и поведение программного продукта. Для таких специалистов дизайн или проектирование интерфейса есть в первую очередь тон, стиль, композиция, которые являются атрибутами бренда, во вторую очередь - прозрачность и понятность информации и лишь затем - передача информации о поведении посредством ожидаемого назначения.

Дизайнерам визуальной части интерфейса необходимы некоторые навыки, которые присущи графическим дизайнерам, но они должны еще обладать глубоким пониманием и правильным восприятием роли поведения. Их усилия в значительной степени сосредоточены на организационных аспектах проектирования. В центре их внимания находится соответствие между визуальной структурой интерфейса с одной стороны и логической структурой пользовательской ментальной модели и поведения программы - с другой. Кроме того, их заботит вопрос о том, как сообщать пользователю о состояниях программы и что делать с когнитивными аспектами пользовательского восприятия функций.

**5.1.3 Визуальный информационный дизайн**

Информационные дизайнеры работают над визуализацией данных, содержимого и средств навигации. Усилия информационного дизайнера направлены на то, чтобы представить данные в форме, способствующей их верному истолкованию. Результат достигается через управление визуальной иерархией при помощи таких средств, как цвет, форма, расположение и масштаб. Распространенными объектами информационного дизайна являются всевозможные графики, диаграммы и прочие способы отображения количественной информации.

Чтобы создавать привлекательные и удобные пользовательские интерфейсы, дизайнер интерфейсов должен владеть базовыми визуальными навыками - пониманием цвета, типографики, формы и композиции - и знать, как их можно эффективно применять для передачи поведения и представления информации, для создания настроения и стимулирования физиологических реакций. Дизайнеру интерфейса также требуется глубокое понимание принципов взаимодействия и идиом интерфейса, определяющих поведение продукта.

**5.2 Строительные блоки визуального дизайна интерфейсов**

*Дизайн* интерфейсов сводится к вопросу о том, как оформить и расположить визуальные элементы таким образом, чтобы внятно отразить поведение и представить информацию. Каждый элемент визуальной композиции имеет ряд свойств, и сочетание этих свойств придает элементу смысл. *Пользователь* получает возможность разобраться в интерфейсе благодаря различным способам приложения этих свойств к каждому из элементов интерфейса. В тех случаях, когда два объекта обладают общими свойствами, *пользователь* предположит, что эти объекты связаны или похожи. Когда пользователи видят, что свойства отличаются, они предполагают, что объекты не связаны.

Создавая пользовательский *интерфейс*, проанализируйте перечисленные ниже визуальные свойства каждого элемента или группы элементов. Чтобы создать полезный и привлекательный пользовательский *интерфейс*, следует тщательно поработать с каждым из этих свойств.

**5.2.1 Форма**

Форма - главный признак сущности объекта для человека. Мы узнаем объекты по контурам. Если мы увидим на картинке синий ананас, мы его сразу опознаем, потому что мы помним его форму. И лишь потом мы удивимся странному цвету (см. [рис. 5.1](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=1#image.5.1)). При этом различение форм требует большей концентрация внимания, чем анализ цвета или размера. Поэтому форма - не лучшее свойство для создания контраста, если требуется привлечь внимание пользователя.



**Рис. 5.1.**В первую очередь мы видим ананас, а уже потом начинаем задумываться, почему он синий

**5.2.2 Размер**

Более крупные элементы привлекают больше внимания, особенно если они значительно превосходят размерами окружающие элементы. Люди автоматически упорядочивают объекты по размеру и склонны оценивать их по размеру; если у нас есть текст в четырех размерах, предполагается, что относительная важность текста растет вместе с размером и что полужирный текст более важен, чем текст с нормальным начертанием. Таким образом, размер - полезное свойство для обозначения информационных иерархий.

**5.2.3 Цвет**

Цветовые различия быстро привлекают внимание. В некоторых профессиональных областях цвета имеют конкретные значения, и этим можно пользоваться. Так, для бухгалтера красный цвет - отрицательные результаты, а черный - положительные.

Цвета приобретают смыслы и благодаря социальным контекстам, в которых проходит наше взросление. Например, белый цвет на Западе ассоциируется с чистотой и миром, а в Азии и арабских странах - с похоронами и смертью. При этом цвет изначально не обладает свойством упорядоченности и не выражается количественно, поэтому далеко не идеален для передачи информации такого рода. Кроме того, не следует делать цвет единственным способом передачи информации, поскольку цветовая слепота встречается довольно часто.

Применяйте цвет с умом. Чтобы создать эффективную визуальную систему, позволяющую пользователю выявлять сходства и различия объектов, используйте ограниченный набор цветов - эффект радуги перегружает восприятие пользователя и ограничивает возможности по передаче ему информации.

Выбор цветовой палитры для программы необходимо проводить очень осторожно. По разным данным, той или иной формой цветовой слепоты страдают до 10% мужчин, и использование, например, красного и зеленого цветов для указания контраста затрудняет работу с приложением для этих людей.

**5.2.4 Яркость**

Понятия темного и светлого обретают смысл преимущественно в контексте яркости фона. На темном фоне темный текст почти не виден, тогда как на светлом он будет резко выделяться. Контрастность люди воспринимают легко и быстро, так что значение яркости может стать хорошим инструментом привлечения внимания к тем элементам, которые требуется подчеркнуть. Значение яркости - также упорядоченная переменная, например, более темные (с более низкой яркостью) цвета на карте легко интерпретируются: они обозначают большие глубины или большие значения других параметров.

**5.2.5 Направление**

Направление полезно, когда требуется передавать информацию об ориентации (вверх или вниз, вперед или назад). Помните, что восприятие направления может быть затруднено в случае некоторых форм и при малых размерах объектов, поэтому ее лучше использовать в качестве вторичного признака. Так, если требуется показать, что рынок акций пошел вниз, можно использовать направленную вниз стрелку красного цвета.

**5.2.6 Текстура**

Разумеется, изображенные на экране элементы не обладают настоящей текстурой, но способны создавать ее видимость. Текстура редко бывает полезна для передачи различий или привлечения внимания, поскольку требует значительной концентрации на деталях. И тем не менее текстура может быть важной подсказкой. Засечки и выпуклости на элементах пользовательского интерфейса обычно указывают, что элемент можно перетаскивать, а фаски или тени у кнопки усиливают ощущение, что ее можно нажать.

**5.2.7 Расположение**

Расположение - это переменная, упорядоченная и выражаемая количественно, а значит, полезная для передачи иерархии. Расположение также может служить средством создания пространственных отношений между объектами на экране и объектами реального мира (например, небо в верхней половине, земля в нижней).

Расположение элементов мобильного приложения очень сильно влияет на удобство использования и зависит от того, как пользователь будет держать устройство (см. [рис. 5.2](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=1#image.5.2)). Подробнее об этом будет рассказано в продолжении данного курса.



**Рис. 5.2.**Различные варианты удержания смартфона

### 5.3 Элементы управления и дизайн навигации

*Элементы управления* - это доступные для манипулирования самодостаточные экранные объекты, посредством которых люди взаимодействуют с цифровыми продуктами. *Элементы управления* (controls, другое название - widgets - сокращение от *windows* gadgets - оконные приспособления) - это базовые строительные блоки графического пользовательского интерфейса.

Рассматривая *элементы управления* с учетом целей пользователя, их можно разбить на четыре основные категории:

* **командные элементы управления**, применяемые для выполнения функций;
* **элементы выбора**, позволяющие выбирать данные или настройки;
* **элементы ввода**, применяемые для ввода данных;
* **элементы отображения**, используемые для наглядного непосредственного манипулирования.

Некоторые *элементы управления* сочетают в себе свойства более чем одной категории.

#### 5.3.1 Командные элементы управления

Командные элементы управления выполняют действия, причем делают это немедленно. Главным и по сути единственным командным элементом является кнопка, которая обладает множеством вариантов отображения. Элементы меню также являются командными идиомами.

##### Кнопки

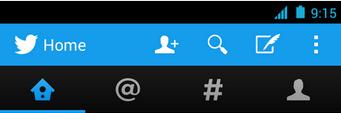
Кнопки обычно легко опознаются благодаря их псевдотрехмерности (рис. 5.3). Действие выполняется сразу после нажатия на кнопку. Часто особым образом выделяется кнопка по умолчанию, соответствующая наиболее часто используемому действию.

|  |  |
| --- | --- |
| http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_03_1.jpg | http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_03_2.jpg |
| **Рис. 5.3** Скриншоты популярной игры "Cut the Rope". Кнопки кажутся выпуклыми, а нажатая кнопка меняет цвет | |

Кнопка - удобный и привлекательный с визуальной точки зрения элемент управления. Весь ее облик подсказывает, что на нее можно нажать, и это характеризует ее ожидаемое назначение. Рекомендуется изменять внешний вид нажатой кнопки, так как это облегчает понимание работы программы пользователем.

##### Кнопки-значки

Кнопки, помещенные на панель инструментов, обычно становятся квадратными, теряют текстовую надпись и обзаводятся пиктограммой - пояснением в виде графического значка ([рис. 5.4](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=2#image.5.4)).



**Рис. 5.4.**Кнопки-значки в Android-приложении Twitter

Считается, что кнопки-значки очень удобны: они постоянно на виду и взаимодействовать с ними проще, чем с элементами раскрывающегося меню. Поскольку они постоянно видны, то легко запоминаются. У большинства пользователей не возникает вопросов относительно ожидаемого назначения кнопки. Проблема в том, что изображение на кнопке иногда бывает непонятным. Например, пиктограмма с изображением дискеты, традиционно обозначающая сохранение, часто непонятна молодым пользователям, которые никогда не работали с реальными дискетами.

##### Гиперссылки

Текстовые гиперссылки - распространенный способ навигации в сети и веб-приложениях, однако при программировании для мобильных устройств их следует избегать. Дело в том, что попасть по ссылке пальцем с первого раза часто затруднительно и пользователей раздражает необходимость повторения этих действий.

#### 5.3.2 Элементы управления выбором

Элементы выбора позволяют пользователю выбрать из группы допустимых объектов тот, с которым будет совершено действие. Элементы выбора применяются также для действий по настройке. Распространенными элементами выбора являются флажки и списки.

Раньше использование элементов управления выбором не приводило к немедленному выполнению действий - требовалась еще и активация командного элемента. Сейчас возможны оба варианта. Если желательно дать пользователю возможность несколько раз осуществить выбор перед выполнением действия, следует создать явный командный элемент управления (кнопку). Если же пользователю полезно сразу видеть результат своих действий, и эти действия легко отменить, разумно сделать так, чтобы элемент выбора играл также и роль командного элемента.

##### Флажки

Назначение флажка очевидно. Щелкнув по флажку, пользователь немедленно увидит появившуюся галочку. Флажок прост, нагляден и изящен, однако основан на тексте. Качественный текст может исключить возможность неоднозначного толкования флажка. Однако этот же поясняющий текст вынуждает пользователя замедляться для прочтения, а также занимает значительное экранное пространство.

Традиционно флажки имеют квадратную форму. Не забывайте, что пользователи распознают визуальные объекты по форме, и квадратная форма флажков - важный стандарт.

##### Выключатели

Существует возможность сделать флажок более наглядным, применив в качестве основы кнопку-значок, которая может фиксироваться в нажатом состоянии. Такой элемент называется **выключателем** ([рис. 5.5](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=3#image.5.5)).



**Рис. 5.5.**Один из самых распространенных выключателей - кнопка "Pause" в играх

Состояние выключателя остается неизменным до следующего щелчка. Выключатели экономно расходуют экранное пространство: они занимают меньше места, потому что их назначение описывается не с помощью текста, а посредством визуальных средств. Разумеется, это означает, что им присущ тот же недостаток обычных кнопок-значков - неоднозначность пиктограмм.

##### Триггеры

Кнопки-триггеры - это разновидность элементов управления. Они призваны экономить экранное пространство, к сожалению, ценой значительной дезориентации пользователя. Классический пример - размещение на одной кнопке функций воспроизведения и паузы для музыкального проигрывателя. Подводным камнем такого подхода является то, что элемент управления можно ошибочно посчитать индикатором состояния проигрывателя ("на паузе" или "идет воспроизведение"). Элемент управления может служить либо индикатором состояния, либо кнопкой переключения состояний, но не тем и другим одновременно ([рис. 5.6](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=3#image.5.6)).

Если кнопка говорит "ВКЛЮЧЕНО", когда находится в состоянии "выключено", то непонятно, в каком же состоянии находится кнопка. Если кнопка говорит "ВЫКЛЮЧЕНО", когда находится в состоянии "выключено", тогда возникает вопрос, где искать кнопку "ВКЛЮЧЕНО"?

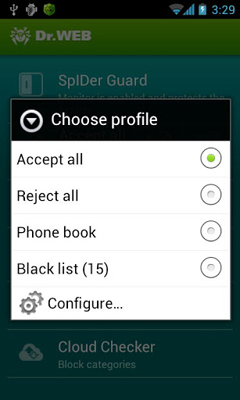
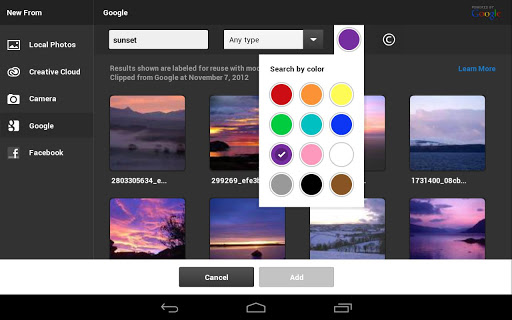
**Рис. 5.6.**Если кнопка говорит "ВКЛЮЧЕНО", когда находится в состоянии "выключено", то непонятно, в каком же состоянии находится кнопка. Если кнопка говорит "ВЫКЛЮЧЕНО", когда находится в состоянии "выключено", тогда возникает вопрос, где искать кнопку "ВКЛЮЧЕНО"?

##### Радиокнопки

Радиокнопки внешне похожи на флажки ([рис. 5.7](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=3#image.5.7)), но являются взаимоисключающими, то есть выбор одного из вариантов автоматически аннулирует предыдущий выбор. В каждый момент времени может быть выбрана только одна кнопка. Радиокнопки всегда объединяются в группы из двух или более радиокнопок, причем в каждой группе одна радиокнопка всегда выбрана. Радиокнопки всегда круглые по той же причине, по которой флажки всегда имеют квадратную форму: именно такими они были изначально.

Радиокнопки занимают даже больше места, чем флажки, однако в некоторых случаях такой расход экранного пространства оправдан.

Кнопка-значок преобразовала радиокнопки так же, как флажки, заменив их в основном интерфейсе приложения. Если два или более выключателя объединены схемой взаимного исключения - так, чтобы в каждый момент мог быть включен лишь один из них, - они ведут себя точно так же, как радиокнопки. Так образуются **радиокнопки со значками**. Элементы управления цветом в Adobe Photoshop - хороший пример радиокнопок со значками ([рис. 5.8](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=3#image.5.8)).

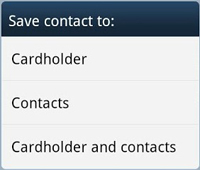
 

**Рис. 5.7.**Радиокнопки **Рис. 5.8.**Элементы управления цветом в Android-приложении Adobe

Android-приложения Dr.Web Photoshop представляют собой групп радиокнопок со значками

##### Списки

Элементы управления типа "список" позволяют осуществлять выбор из конечного множества текстовых строк, каждая из которых представляет команду, объект или признак. Подобно радиокнопкам, списки - мощный инструмент, упрощающий взаимодействие за счет устранения возможности неправильного выбора. Списки - это небольшие текстовые области с полосой прокрутки, автоматически подключаемой при необходимости ([рис. 5.9](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=3#image.5.9)). Пользователь может выбрать единственную строку текста, нажав на нее.



**Рис. 5.9.**Стандартный список в Android

**Раскрывающийся список** - повсеместно встречающийся вариант обычного списка. Он показывает лишь выбранный элемент в одну строку, но если нажать на стрелку, открываются другие варианты выбора.

Элемент управления **представление в виде списка** предоставляет возможность сопровождать каждую строку текста пиктограммой. Такая возможность весьма полезна - существует множество ситуаций, когда можно упростить работу пользователя, располагая графические идентификаторы рядом со строками важных вариантов выбора ([рис. 5.10](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=3#image.5.10)).

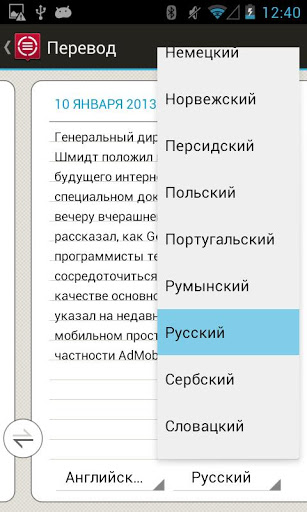


**Рис. 5.10.**Элемент управления типа "список" с пиктограммами в Android-приложении, позволяющий визуально оценивать погоду в различных городах

##### Комбо-списки и комбо-кнопки

Комбо-элементы представляют собой сочетание элементов. Комбо-кнопка - разновидность радиокнопки со значком. Обычно она выглядит как кнопка-значок с небольшой стрелкой, но если нажать на стрелку и удерживать ее в нажатом состоянии, разворачивается меню.

Комбо-список представляет собой сочетание списка и поля редактирования ([рис. 5.11](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=3#image.5.11)).



**Рис. 5.11.**Раскрывающееся поле со списком в Android-приложении ABBYY Translator позволяет выбрать нужный язык из списка

Вариант с раскрывающимся списком значительно экономит экранное пространство. Комбо-список хорошо подходит для тех случаев, когда необходимо организовать выбор единственного объекта.

#### 5.3.3 Элементы ввода

Элементы ввода дают пользователю возможность не только выбирать существующие сведения, но и вводить новую информацию. Самый простой элемент - поле редактирования текста (поле ввода). В эту категорию попадают также такие элементы управления, как счетчики и ползунки.

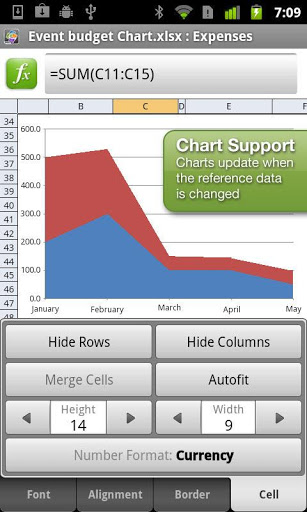
##### Ограничивающие элементы ввода

Любой элемент управления, ограничивающий набор значений, доступных для ввода пользователем, является ограничивающим элементом ввода. Так, например, ползунок со шкалой значений от 0 до 100 является ограничивающим элементом ввода. Независимо от действий пользователя не может быть введено число, выходящее за диапазон определенных программой значений. Проще говоря, ограничивающие элементы ввода должны использоваться везде, где необходимо ограничить множество допустимых значений.

Ограничивающий элемент ввода должен четко информировать пользователя о допустимых границах. Текстовое поле, которое отвергает ввод пользователя после того, как он выполнил ввод, не может считаться ограничивающим элементом управления. Если пользователь должен выразить выбор числовым значением в определенных границах, предоставьте ему элемент управления, сообщающий об этих границах и предотвращающий ввод недопустимых значений. Такую возможность дает ползунок. Ползунок позволяет пользователю определять числовые значения в относительных терминах, а не в результате непосредственного ввода с клавиатуры. Но для ввода точных значений лучше подходят счетчики.

##### Счетчики

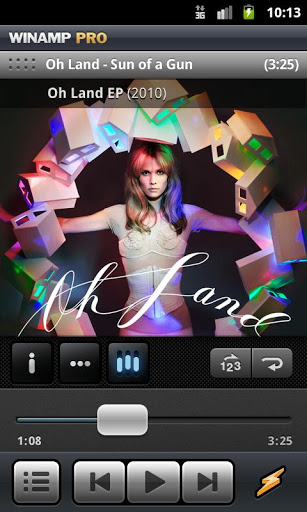
Счетчик состоит из небольшого поля ввода и двух прикрепленных к нему кнопок ([рис. 5.12](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=4#image.5.12)). Благодаря счетчикам грань между ограничивающими и неограничивающими элементами ввода данных становится размытой. Маленькие кнопки со стрелками позволяют пользователю изменять значение в поле редактирования небольшими шагами. Эти шаги могут выполняться до определенного предела: значение не может превысить максимум, установленный программой, или стать меньше установленного минимума. Если пользователь пожелает ввести определенное число, он может сделать это за счет прямого ввода числа в поле редактирования.



**Рис. 5.12.**Реализация счетчиков в Android-приложении Quickoffice

##### Рукоятки и ползунки

Рукоятки и ползунки очень эффективно расходуют экранное пространство, и оба этих элемента управления замечательно справляются с задачей обеспечения визуальной обратной связи по настройкам ([рис. 5.13](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=4#image.5.13)). Ползунки и рукоятки применяются в основном в качестве ограничивающих элементов управления ввода. Например, ползунки - превосходное средство для действий, связанных с масштабированием.



**Рис. 5.13.**Ползунок в Android-приложении Winamp осуществляет перемотку воспроизводимой композиции

##### Неограничивающие элементы ввода

Пожалуй, главный неограничивающий элемент ввода - поле ввода текста. Этот простейший элемент управления позволяет пользователям набирать любые алфавитно-цифровые строки. Как правило, поля ввода - это небольшие области, внутри которых можно набрать одно-два слова, но они могут быть реализованы и в виде довольно сложных текстовых редакторов.

Когда пользователю предложено неограничивающее текстовое поле ввода, которое при этом принимает лишь строки определенного формата, вероятно, имеется необходимость помочь пользователям вводить "допустимые" строки. Имеется множество стандартных форматов вводимых данных - даты, телефонные номера, почтовые индексы, номера социального страхования. Ключ к успешному проектированию элемента ввода с проверкой данных - в хорошо развитой обратной связи с пользователем.

#### 5.3.4 Элементы управления отображением

Элементы управления отображением используются для управления визуальным представлением информации на экране. Типичными примерами элементов отображения являются разделители и полосы прокрутки. Сюда же входят разделители страниц, линейки, направляющие, сетки и рамки.

##### Текстовые элементы

Вероятно, самый простой элемент управления отображением - элемент вывода текстовой информации, который отображает текстовое сообщение в некоторой позиции на экране. Он предоставляет текстовые метки для других элементов управления и выводит данные, которые не могут или не должны быть изменены пользователем. Единственная серьезная проблема этого элемента состоит в том, что он зачастую используется там, где должны присутствовать элементы ввода (и наоборот).

##### Полосы прокрутки

Полосы прокрутки служат важной цели - они позволяют осмысленным образом помещать большие объемы информации внутри рамок окон и панелей. К сожалению, они расходуют экранное пространство и ими сложно манипулировать. Однако замечательное преимущество полосы прокрутки состоит в создании контекста текущего положения в окне. Бегунок полосы прокрутки указывает текущее положение и нередко масштаб "территории", доступной для прокрутки.

##### Разделители

Разделители - удобный инструмент для разделения главного окна приложения на несколько связанных между собой панелей, в каждой из которых можно просматривать, изменять или переносить ту или иную информацию. Подвижные разделители всегда должны сообщать о своей подвижности посредством изменения формы курсора. Однако следует проявлять осторожность, выбирая, какие именно разделители должны стать подвижными. В общем случае разделитель не должен перемещаться таким образом, чтобы содержимое панели становилось непригодным к использованию.

##### Выдвижные панели

Выдвижные панели - это панели приложения, которые можно открывать и закрывать в одно действие. Выдвижные панели - замечательное место для элементов управления и функций, которые используются совместно с основной рабочей областью приложения, но не столь часто. Выдвижные панели более удобны, чем диалоговые окна, так как не закрывают основное окно (рис. 5.14).

|  |  |
| --- | --- |
| http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_14_1.jpg | http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_14_2.jpg |
| **Рис. 5.14.** Скриншоты популярной игры "Cut the Rope". Новостная панель появляется, если потянуть за кольцо | |

### 5.4 Рекомендации по проектированию GUI под Android

#### 5.4.1 Рекомендации разработчиков. Android Guideline

Когда платформа Android только появилась, не было никаких рекомендаций по разработке дизайна, поэтому все разработчики проектировали внешний вид приложений по своему вкусу. Отсутствие единого стиля сказалось на интерфейсах не лучшим образом, многие программы были откровенно некрасивы и неудобны. Кроме того, операционная система Android работает на устройствах с различными экранами, и разработчику необходимо помнить, что его приложение должно масштабироваться под различные параметры смартфонов и планшетов.

В настоящее время существует стандарт Android Design, и, если вы хотите, чтобы ваше приложение стало по-настоящему популярным и нужным, настоятельно рекомендуем его придерживаться. Далее мы рассмотрим основные принципы дизайна. Разумеется, в рамках этого курса невозможно учесть все нюансы. В списке источников есть ссылка на рекомендации от Android User Experience Team, к сожалению, все на английском языке.

Приведем выдержки из рекомендаций по дизайну:

* Реальные объекты гораздо веселее, чем кнопки и меню. Позвольте людям манипулировать знакомыми вещами! Тогда работа будет эффективнее.
* Картинки работают быстрее, чем слова.
* Используйте короткие фразы, состоящие из простых слов. Люди часто пропускают предложения, если они слишком длинные.
* Никогда не теряйте пользовательскую информацию. Если человеку придется вводить данные повторно, велика вероятность того, что он откажется использовать ваше приложение.
* Если объекты похожи, они должны выполнять сходные действия.
* Показывайте только то, что необходимо пользователю именно в этот момент.
* Выводите пользователю сообщения, только если вопрос действительно важен.
* Делайте важные вещи быстро.
* Разбивайте сложные задачи на несколько простых шагов.
* Будьте вежливы и корректны в общении с пользователем.
* Пользователь всегда должен быть уверен в том, что он знает, где сейчас находится. На любом шаге он должен иметь возможность вернуться назад, даже если это прервет выполнение какой-то задачи.
* Используйте интерфейсные элементы, которые будут работать в любой ситуации.
* Самый главный принцип - НЕ УСЛОЖНЯЙТЕ пользователю жизнь!

#### 5.4.2 Обзор интерфейса

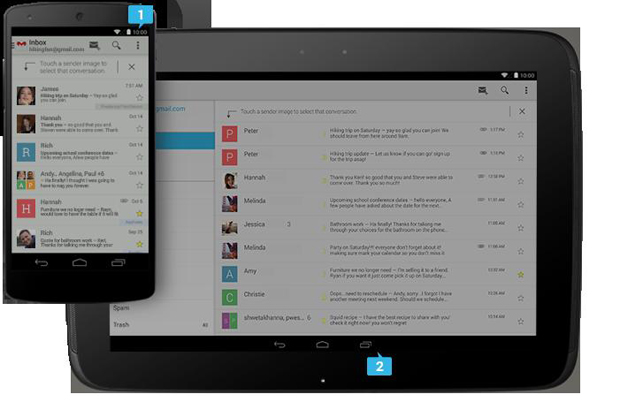
Приведем выдержки из рекомендаций по дизайну приложений для Android. Сделаем краткий обзор интерфейса операционной системы.

Домашний экран - это настраиваемая пользователем область, которая может содержать иконки приложений, папки и виджеты. Смартфон может иметь несколько домашних экранов, навигация между ними осуществляется с помощью перелистывания влево или вправо.

На домашнем экране в центре нижней части есть кнопка для открытия экрана приложений. Экран приложений позволяет пользователю запустить любую из установленных программ. Если устройство было использовано для отладки в процессе разработки, то приложение тоже окажется в этом списке и его можно будет вызвать даже после отключения от компьютера. Если приложение было использовано недавно, его можно найти в списке недавно использованных приложений, который вызывается нажатием на третью кнопку на панели внизу (см. рис. 5.15).

|  |  |  |
| --- | --- | --- |
| http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_15_1.jpg | http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_15_2.jpg | http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_15_3.jpg |
| **Рис. 5.15.** Домашний экран, экран всех приложений и список недавно использовавшихся приложений | | |

В нижней и верхней частях экрана находятся системные панели, предназначенные для размещения уведомлений и навигации по устройству. Нижняя панель (Navigation Bar) предназначена для навигации на тех устройствах, которые не имеют аппаратных навигационных клавиш (все современные устройства). Верхняя часть экрана (Status Bar) предназначена для вывода различных сведений, например, времени, уровня заряда батареи, сигнала сотовой сети, а так же информационных сообщений.

[](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_16.jpg)

[увеличить изображение](http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_16.jpg)  
**Рис. 5.16.**Информационная (1) и навигационная (2) панели

Уведомления - это быстрые сообщения, к которым пользователь может получить доступ в любое время из информационной панели. Это могут быть сообщения об обновлениях или других важных сообщениях, которые не настолько серьезны, чтобы прерывать работу пользователя. Но к ним легко можно получить доступ, потянув вниз верхнюю панель. Нажатие на уведомление вызывает соответствующее сообщение.

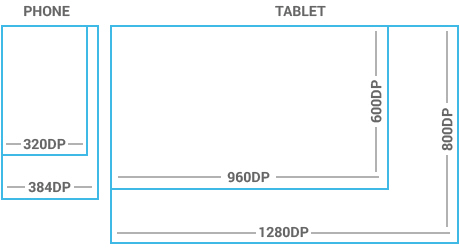
#### 5.4.3 Шрифты

В дизайне Android используются традиционные типографические инструменты, такие как масштаб, разреженность и выравнивание по сетке. Успешное применение этих выразительных средств помогает пользователю воспринимать информацию быстрее. В версии Android 4.0 Ice Cream Sandwich была представлена шрифтовая гарнитура без засечек Roboto, специально разработанная для экранов с высоким разрешением. Набор шрифтов доступен для бесплатной загрузки. Гарнитура включает в себя прямое и наклонное начертания для шрифтов различной ширины (см. рис. 5.17).

|  |  |
| --- | --- |
| http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_17_1.jpg | http://www.intuit.ru/EDI/14_06_16_2/1465856498-31733/tutorial/1258/objects/5/files/05_17_2.jpg |
| **Рис. 5.17.** Шрифт Roboto и его возможные варианты | |

#### 5.4.4 Масштабирование

Устройства различаются не только физическими размерами. Важным параметром является плотность экрана (DPI - количество точек на дюйм). Выделяют несколько категорий плотности экрана для Android-устройств: LDPI, MDPI, HDPI, XHDPI, XXHDPI, и XXXHDPI. Чтобы элементы интерфейса имели одинаковый физический размер на экранах разных устройств, компания Google ввела абстрактную единицу измерения - DP (независимый от разрешения пиксель). Один DP равен одному пикселю на экране типа MDPI. Устройства, имеющие меньше 600dp по короткой стороне, считаются телефонами, в противном случае мы говорим о планшетах (см. [рис. 5.18](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=6#image.5.18)).

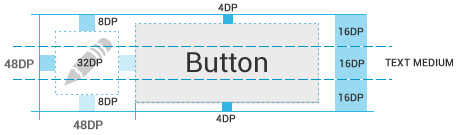


**Рис. 5.18.**Размеры экранов телефонов и планшетов

Соответствие размеров экранов и их плотностей представлено в [таблице 5.1](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=6#table.5.1):

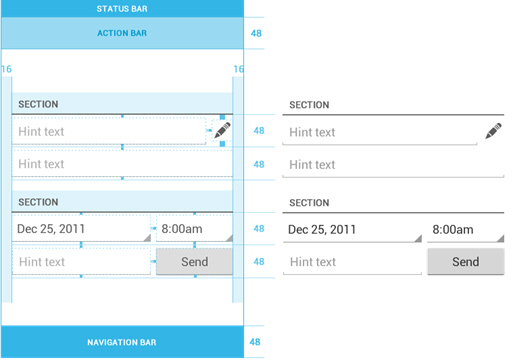
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Таблица 5.1. Плотности и размеры экранов | | | | |
| **№** | **Обозначение** | **Название** | **Соответствие** | **1 dp =** |
| 1 | LDPI | Low density | 120 dpi | 0,75 пикселя |
| 2 | MDPI | Medium density | 160 dpi | 1 пиксель |
| 3 | HDPI | High density | 240 dpi | 1,5 пикселя |
| 4 | XHDPI | Extra-high density | 320 dpi | 2 пикселя |
| 5 | XXHDPI | Extra-extra!-high density | 480 dpi | 3 пикселя |
| 6 | XXXHDPI | Extra-extra-extra!-high density | 640 dpi | 4 пикселя |

Минимальный размер элемента управления - 48dp. Такое значение обусловлено тем, что на реальном устройстве оно соответствует 7-10 миллиметрам. При управлении кончиками пальцев такой размер является минимальным для отделения нужного элемента от всех остальных. Если какой-то из размеров элемента управления должен быть больше, чем 48dp, рекомендуется делать его размеры кратным этому значению (см. [рис. 5.19](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=6#image.5.19)).



**Рис. 5.19.**Размеры элемента управления кратны 48dp

Расстояние между элементами управления рекомендуется делать кратным 8dp (см. [рис. 5.20](http://www.intuit.ru/studies/courses/12643/1191/lecture/21986?page=6#image.5.20)).



**Рис. 5.20.**Пример расположения элементов управления

Прочие нюансы дизайна для Android рассмотрены в лабораторной работе.