Programming Handheld Systems

Adam Porter
User Interface Classes
Today’s Topics

Views & View Events
View Groups, AdapterViews & Layouts
Menus & ActionBar
Dialogs
Android User Interfaces

Activities usually display a user interface

Android provides many classes for constructing user interfaces
View

Key building block for UI components
Occupy a rectangular space on screen
Responsible for drawing themselves and for handling events
Some Predefined Views

Button
ToggleButton
Checkbox
RatingBar
AutoCompleteTextView
Button

View that can be clicked on to perform an action
UIButton
ToggleButton

A 2-state Button
checked/not checked state
Light indicator showing state
UIToggleButton
CHECKBOX

Another kind of 2-state button
Checked/not checked
UICheckbox
RatingBar

A view comprising a row of stars
the user can click or drag the stars to
highlight some number of them
UIRatingBar
**AutoCompleteTextView**

An editable text field that provides completion suggestions as the user types in text.
UIAutoComplete
TextView
**Common View operations**

Set visibility: show or hide view

Set checked state

Set Listeners: Code that should be executed when specific events occur

Set properties: opacity, background, rotation

Manage input focus: allow view to take focus, request focus
View Event Sources

User Interaction
  Touch
  Keyboard/trackball/D-pad

System Control
  Lifecycle changes
Handling View Events

Often handle events with listeners
Numerous Listener interfaces defined by the View class
**View Listener interfaces**

`OnClickListener.onClick()`

*View has been clicked*

`OnLongClickListener.onLongClick()`

*View has been pressed & held*
View Listener interfaces

OnFocusChangeListener.onFocusChange()

View has received or lost focus

OnKeyListener.onKey()

View about to receive a hardware key press
Displaying Views

Views are organized in a Tree
Displaying has multiple steps
Measure – get dimensions of each View
Layout – Position each View
Draw – Draw each view
Handling View Events

Custom View subclasses can override various View methods
Handling View Events

onMeasure()

Determine the size of this View and its children

onLayout()

View must assign a size and position to all its children

onDraw()

View should render its content
Handling View Events

onFocusChanged()

View’s focus state has changed

onKeyUp(), onKeyDown()

A hardware key event has occurred

onWindowVisibilityChanged()

Window containing view has changed its visibility status
**ViewGroup**

An invisible View that contains other views

Used for grouping & organizing a set of views

Base class for view containers & layouts
Some Predefined ViewGroups

RadioGroup
TimePicker
DatePicker
WebView
MapView
Gallery
Spinner
RadioGroup

A ViewGroup containing a set of Radio Buttons (CheckBoxes)

Only one button can be selected at any one instant
UIRadioGroup
**TimePicker**

A ViewGroup that allows the user to select a time
**DatePicker**

A **ViewGroup** that allows the user to select a date
UIDatePicker
**WebView**

A `ViewGroup` that displays a web page
UIWebView
MapView

A ViewGroup that displays a Map
UI Google Maps
Adapters & AdapterViews

AdapterViews are views whose children and managed by an Adapter. Adapter manages the data and provides data views to AdapterView. AdapterView displays the data Views.
**ListView**

**AdapterView** displaying a scrollable list of selectable items

Items managed by a **ListAdapter**

**ListView** can filter the list of items based on text input
UIListView
Spinner

An AdapterView that provides a scrollable list of items
User can select one item from the list
Items managed by a Spinner Adapter
UISpinner
**Gallery**

A **ViewGroup** showing a horizontally scrolling list

**Items managed by a SpinnerAdapter**
UIGallery
Layouts

A generic Viewgroup that defines a structure for the Views it contains
LinearLayout

Child views arranged in a single horizontal or vertical row
UILinearLayout
[2013-05-28 13:50:42 - UILinearLayout] ActivityManager: Starting: Intent { act=android.intent.action.MAIN flg=0x00000080 cmp=course.examples.UI.LinearLayout/MainActivity class=course.examples.UI.LinearLayout.MainActivity }
RelativeLayout

Child views are positioned relative to each other and to parent view
TableLayout

Child views arranged into rows & columns
UITableLayout
GridView

Child views arranged in a two-dimensional, scrollable grid
Menus and Action Bar

Activities support menus
Activities can
  Add items to a menu
  Handle clicks on the menu items
Menu Types

Options

Menu shown when user presses the menu button

Context

View-specific menu shown when user touches and holds the view

Submenu

A menu activated when user touches a visible menu item
Options Menus
Context Menus
SubMenus
Creating Menus

Define menu resource in XML file

Store in res/menu/filename.xml
Creating Menus

Inflate menu resource using MenuInflater in onCreate...Menu() methods

Handling item selection in appropriate on...ItemsSelected() methods
Hello Android
With Menus
Menus

Many other features supported

Grouping menu items

Binding shortcut keys to menu items

Binding Intents to menu items
ActionBar

Similar to Application Bar in many desktop applications
Enables quick Access to common operations
FragmentDynamicLayout WithActionBar

Shows play titles and one quote from the selected play

Provides actions for the ActionBar

Three main objects

QuoteViewerActivity

TitleFragment

QuoteFragment
ActionBar.Tab

Screen is divided into tab & content areas

Allows multiple Fragments to share single content area
**ActionBar.Tab**

Each tab is associated with one Fragment

Exactly one tab is selected at any given time

Fragment corresponding to the selected tab is visible in the content area
UITabLayout
Dialogs

Independent subwindows used by Activities to communicate with user
Dialog subclasses

AlertDialog
ProgressDialog
DatePickerDialog
TimePickerDialog
UIAlertDialog
ProgressDialog
Next Time

User Notifications