

1. Record Nr.	UNINA9910300474403321
Autore	Jackson Wallace
Titolo	Pro Android UI // by Wallace Jackson
Pubbl/distr/stampa	Berkeley, CA : , : Apress : , : Imprint : Apress, , 2014
ISBN	1-4302-4987-0
Edizione	[1st ed. 2014.]
Descrizione fisica	1 online resource (565 p.)
Disciplina	004 005.258 005.437 4019
Soggetti	Mobile computing User interfaces (Computer systems) Mobile Computing User Interfaces and Human Computer Interaction
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	"Customizing Unique Android UI Designs"--Cover. Includes index.
Nota di contenuto	""Contents at a Glance""; ""Contents""; ""About the Author""; ""About the Technical Reviewer""; ""Acknowledgments""; ""Part 1: Introduction to the Core Classes for Android UI Design: Development Tools, Layout Containers and Widgets""; ""Chapter 1: Android UI Design Tools: Setting Up Your Android Development System""; ""Java 6: Download and Install the Foundation for Android""; ""Android ADT Bundle: Find and Download Androida€™s IDE""; ""Android ADT Bundle: Installing the Android IDE""; ""Android ADT Bundle: Launch the IDE and Update the ADT"" ""UI Wireframing Tool: Downloading and Installing Pencil""""Digital Image Editing: Download and Install GIMP 2.8.10""; ""3D Modeling and Animation: Download and Install Blender""; ""Digital Audio Editing: Downl oad and Install Audacity 2.0.5""; ""Digital Video Editing: Download and Install Lightworks 11.5""; ""Summary""; ""Chapter 2: Android UI Layouts: Layout Containers and the ViewGroup Class""; ""Layout Containers: Using XML for Tag-based UI Design""; ""Using Parameters: Refining and Customizing the UI Tags""; ""Android

ViewGroup: Our Layout Container's Superclass

ViewPager.LayoutParams: Nested Class Controls Layout

MarginLayoutParams: Nested Class for Margin

The Difference Between Margins and Padding Parameters

Animating Your ViewGroup: LayoutAnimationController

Create a User Interface Project and pro.android.ui Package

The New Android Application Dialog

New Android Application - Configure Project Dialog

New Android Application - Configure Launcher Icon

New Android Application - Create Activity Dialog

New Android Application - Blank Activity Dialog

ViewPager XML Attributes: Configuring the RelativeLayout

View XML Attributes: Configuring the TextView UI Widget

View Padding Parameter: Add Space Inside the Container

The AnalogClock Widget: Adding a View to the ViewGroup

Using the Values Folder: Finish Your UI Design Using XML

Summary

Chapter 3: Android UI Widgets: User Interface Widgets and the View Class

The Android View Class: The Superclass for UI Widgets

Android View Direct Subclasses: Key Widgets and Utilities

Using View Properties: Important Parameters

Focus of a View: Which UI Widget has the User's Attention

View Listeners: Making a UI Widget Respond to the User

UI Layout Creation: Using the setContentView() Method

Adding a Button UI Widget to Our RelativeLayout Container

Eclipse ADT Problems Tab: Debugging XML Errors in ADT

Relative Positioning: Adding the UI Button to the Hierarchy

String Constants: Adding a Tag for Our UI Button

Event Handling: Adding an OnClickListener in Java Code

Controlling AnalogClock Visibility: .setVisibility() Method

Install and Configure an Emulator: Testing Our Java Code

Sommario/riassunto

If you're an Android application developer, chances are you're using fixed, scrolling, swipe-able, and other cutting-edge custom UI Designs in your Android development projects. These UI Design approaches as well as other Android ViewGroup UI layout containers are the bread and butter of Pro Android User Interface (UI) design and Android User Experience (UX) design and development. Using a top down approach, Pro Android UI shows you how to design and develop the best user interface for your app, while taking into account the varying device form factors in the increasingly fragmented Android environment. Pro Android UI aims to be the ultimate reference and customization cookbook for your Android UI Design, and as such will be useful to experienced developers as well as beginners. With Android's powerful UI layout classes, you can easily create everything from the simplest of lists to fully tricked-out user interfaces. While using these UI classes for boring, standard user interfaces can be quite simple, customizing a unique UI design can often become extremely challenging.
