

Compartir

5

Más

Siguiente blog>

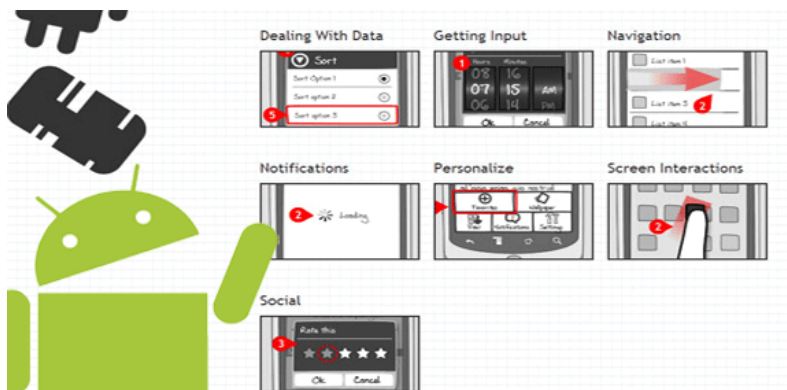
Crear un blog Acceder



--^^-- Mucho Android --^^-- Programacion Java para Android. Aprende sobre este nuevo Sistema Operativo que llego para quedarse con nosotros. Este blog es para estar al tanto de lo ultimo referente al sistema operativo Android.

DOMINGO, 29 DE ABRIL DE 2012

Diseñar Apps Android, buenas practicas y patrones

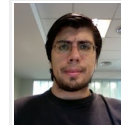


Para terminar el mes en el Blog, quiero charlar sobre una de las cosas más fascinantes en el desarrollo de Android, que es sin duda, el diseño de las pantallas que invitarán al usuario a usar tu aplicación. Si bien, la funcionalidad no deja de ser importante, en el mundo móvil he observado el patrón constante de que los usuarios no sólo buscan que la aplicación haga lo que tenga que hacer sino que además las pantallas con las que interactúan sean llamativas, bonitas y usables.

Navegando en la web, me he encontrado con un excelente artículo titulado "Designing for Android" de [Dan McKenzie](#), que me pareció muy interesante y completo y cuya traducción al español vale la pena para que más personas puedan utilizarlo como referencia. Sin más les dejo mi interpretación del artículo:

Para los diseñadores, Android es el referente a tomar en cuenta cuando se habla de diseño de aplicaciones. Por mucho que a los diseñadores les gustaría pensar que se trata de un mundo de iOS, en el que todos los usuarios utilizáramos iPhone, iPad y la

DATOS PERSONALES



[Lic. Mariano Salvetti](#)

Born in Canada de Gomez (Santa Fe - Argentina), since 2002 I live in Rosario, where he studied at the National University of Rosario's career degree in Computer Science. The rest of my life? I'm nerd !

[Ver todo mi perfil](#)

TE NOTIFICO LOS NUEVOS POST POR EMAIL

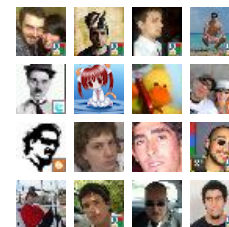
Submit

SEGUIDORES

Participar en este sitio

Google Friend Connect

Miembros (23) [Más »](#)



¿Ya eres miembro? [Acceder](#)

ARCHIVO DEL BLOG

- ▼ 2012 (200)
 - septiembre (1)
 - julio (21)

App Store, nadie puede ignorar que Android tiene actualmente la mayor parte de la cuota de mercado de los smartphones y que su uso se está extendiendo hacia todo tipo de dispositivos electrónicos. En poco tiempo, la plataforma Android de Google ha crecido rápidamente y las distintas marcas la han empezado a notar.

Pero seamos realistas. Los múltiples dispositivos con Android que podemos encontrar en el mercado hacen sentir que el diseño para esta plataforma es una cuestión difícil de dominar. Sumando además que la poca documentación es difícilmente un buen punto de partida para empezar a diseñar y producir aplicaciones que luzcan realmente geniales. Navegando por la web en busca de recursos que te hablen del diseño en Android te hará darte cuenta de que sólo encontrarás contados recursos al respecto y que poco de ellos te servirán realmente como guía.

Si esto te parece un poco desalentador (y es la razón por la que no te has animado a diseñar aplicaciones para Android) te tengo una buena noticia, no estás solo. Afortunadamente, Android está comenzando a abordar los problemas de tener varios dispositivos y tamaños de pantalla, y los fabricantes de dispositivos están adoptando lentamente los estándares que eventualmente reducirán la complejidad en el tema de diseño.

Este artículo les ayudará a los diseñadores a familiarizarse con lo que necesitan saber para empezar a trabajar con Android y para entregar los resultados adecuados para el equipo de desarrollo. Los temas que vamos a cubrir son los siguientes:

- Disminuir la densidad de Android pantalla.
- Conocer los aspectos fundamentales del diseño de Android a través de patrones de diseño.
- Diseñar los requerimientos a partir de las necesidades de desarrollo.
- Qué es Android 3.x y qué hay que conocer

Smartphones con Android y tamaños de pantalla

Al comenzar cualquier proyecto de diseño digital, la comprensión del hardware como primer paso es una buena idea. Para aplicaciones de iOS, correspondería al iPhone y al iPod Touch. En Android, por su parte, esto se extiende a docenas de dispositivos y fabricantes. ¿Por dónde empezar?

La línea de tiempo de las pantallas soportadas en los dispositivos Android comienza por el T-Mobile G1, el primer dispositivo Android disponible en el mercado que tiene una pantalla HVGA de 320 x 480 píxeles.

HVGA significa "la mitad de la matriz de gráficos de video" (o VGA de medio tamaño) que es el tamaño de pantalla estándar para los smartphones de hoy en día. El iPhone 3Gs, 3G y 2G utilizan la misma configuración.

Para simplificar las cosas, Android clasifica las pantallas (tomando en cuenta la longitud de una diagonal que va desde la esquina superior izquierda a la esquina inferior derecha de la pantalla del dispositivo) en cuatro tamaños generales: pequeño, normal, grande y extra grande.

► junio (19)

► mayo (25)

▼ abril (35)

[Introducción a ActionBar Sherlock](#)

[Diseñar Apps Android, buenas practicas y patrones](#)

[Trabajando con la documentación de Android](#)

[Testing en Android](#)

[Componentes de una aplicación Android](#)

[Google Drive, está aca!](#)

[Mi primer proyecto con Maven en Eclipse \(sin depen...](#)

[Instalacion de Maven 2.x y del plugin m2e para Ecl...](#)

[Primeros pasos con GreenDroid](#)

[Guía: Ice Cream Sandwich para desarrolladores](#)

[Android o iPhone, la gran elección](#)

[Quién usa y cómo usa su Android](#)

[El Nexus One fracasó, Android no](#)

[Android 2.3, Características para usuarios](#)

[Funciones de Android 2.3 para desarrolladores](#)

[Guía para iniciales en Android](#)

[Cocos2d para Android, el HolaMundo!](#)

[kit de desarrollo de Android - versión 18](#)

[Hoy Tesis de Licenciatura en Matematica](#)

[Curso Android Rosario](#)

[HOY: Lanzamiento GTUG Rosario y Argentina 2012](#)

[Ice Cream Sandwich crece poco a poco.](#)

[Cronómetros y relojes en Aplicaciones Android](#)

[El oficio de desarrollar apps en la Argentina](#)

[Instagram para Android en la Google Play Store](#)

[Trabajando con el Acelerometro](#)

[GTUG Lanzamiento Argentina 2012](#)

[ANDROID, GLOSARIO DE TERMINOS](#)

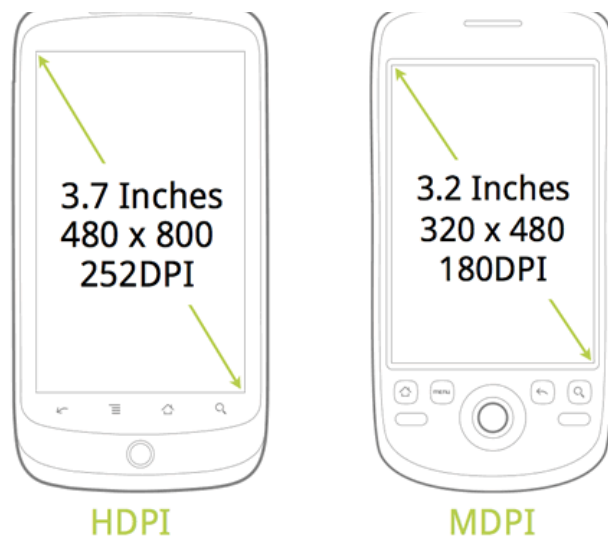
[Programación Android - el "Hola Mundo" de android](#)

[Vamos a Programar en Android](#)

[10 aplicaciones con estilo Android 4.0 Ice Cream S...](#)

[Cosas a saber si empiezas en Android](#)

[China lidera en dispositivos Android y Apple, EE.U...](#)



Samsung Galaxy Beam:
Teléfono con proyector
incorp...

Android, al día

- [marzo](#) (35)
- [febrero](#) (33)
- [enero](#) (31)
- [2011](#) (89)

Imagen tomada del artículo original en el que se muestran dos tamaños de pantalla muy comunes en Android.

320 × 480 se considera un tamaño "normal" de pantalla en Android. Cuando hablemos de tamaño "extra grande" piensa en las pantallas de las tablets. Sin embargo, hay que tomar en cuenta que los smartphones más populares de hoy en día tienen WVGA (es decir, un VGA más grande) 800+ x 480 píxeles HD de pantalla. Por lo tanto, lo que es "normal" está cambiando rápidamente. Por ahora, vamos a tomar como referente que la mayoría de los smartphones con Android tienen pantallas grandes.

	Low density (120), <i>ldpi</i>	Medium density (160), <i>mdpi</i>	High density (240), <i>hdpi</i>	Extra high density (320), <i>xhdpi</i>
Small screen	QVGA (240x320)		480x640	
Normal screen	WQVGA400 (240x400) WQVGA (240x432)	HVGA (320x480)	WVGA (480x800) WVGA854 (480x854) 600x1024	640x960
Large screen	WVGA800 (480x800) WVGA854 (400x854)	WVGA800 (480x800) WVGA854 (480x854) 600x1024		
Extra large screen	1024x600	WXGA (1280x800) 1024x768 1280x768	1536x1152 1920x1152 1920x1200	2048x1536 2560x1536 2560x1600

Imagen tomada del artículo original cuya tabla muestra las diferentes configuraciones de pantalla disponibles en los skins del emulador del Android SDK.

La variedad de tamaños de pantalla puede ser un reto para los diseñadores que están

tratando de crear un tamaño único para todos los diseños de layout. Para resolver esto, he encontrado que el mejor enfoque consiste en diseñar un conjunto de layouts para 320 x 533 píxeles físicos y luego introducir diseños personalizados para los otros tamaños de pantalla.

Si bien esto crea más trabajo tanto para el diseñador y el desarrollador, el tamaño de pantalla más grande de los dispositivos tales como el Motorola Droid y Evo HTC puede requerir cambios en los diseños base que hagan un mejor uso de los recursos visuales del dispositivo.

¿Qué necesitas saber acerca de las densidades de pantalla?

Los tamaños de pantalla es sólo la mitad del asunto. Los desarrolladores no hacen referencia a la resolución de las pantallas, sino más bien a su densidad. Así es como Android define los términos y conceptos acerca del soporte de las pantallas en la [documentación de Android developers](#):

- Resolución. El número total de píxeles físicos en una pantalla.
- Densidad de la pantalla. La cantidad de píxeles en un área física de la pantalla, normalmente se conoce como DPI (puntos por pulgada).
- Píxeles independientes de la densidad (DP). Esta es una unidad de píxel virtual que se utiliza en la definición de un diseño de interfaz de usuario con el fin de expresar las dimensiones del diseño o la posición de una manera independiente de la densidad. Los píxeles independientes de la densidad equivalen a un píxel físico en una pantalla de 160 DPI, que es la densidad de referencia asumida por el sistema como la densidad "media" de la pantalla. En tiempo de ejecución, el sistema maneja de manera transparente cualquier ampliación de las unidades de DP según sea necesario, en base a la densidad real de la pantalla que se esté utilizando. La conversión de las unidades de DP a los píxeles de la pantalla es simple: los píxeles = DP * (DPI / 160). Por ejemplo, en una pantalla de 240 DPI, un DP es igual a 1,5 píxeles físicos. Utiliza siempre los DP como unidad para definir los diseños de tu interfaz de usuario para asegurarte de que se mostrará correctamente en pantallas con diferentes densidades.

Es un poco confuso, pero esto es lo que necesitas saber: Al igual que los tamaños de pantalla, Android divide a las densidades de pantalla en cuatro densidades básicas: LDPI (bajo), MDPI (medio), HDPI (alto), y XHDPI (muy alta). Esto es importante porque tendrás que entregar todos los elementos gráficos (bitmaps) en sets de diferentes densidades. Por lo menos, tendrás que entregar un set en MDPI y HDPI para cualquier aplicación Android que hagas.

Lo que esto significa es que todos los gráficos de bitmaps necesitan ser ampliados o reducidos partiendo de su tamaño base (320 x 533). Nota: hay también una forma de [parsear los archivos SVG](#) que proporciona una forma de escalar el vector en diferentes tamaños de pantallas y densidades sin perder calidad en la imagen.

El requerimiento de los bitmaps es similar al de preparar gráficos para imprimir contra los que tenemos para la web. Si tienes experiencia en el campo de las impresiones, sabrás que una imagen de 72 PPI se verá muy pixelada y borrosa cuando son ampliadas o impresas. En lugar de ello, necesitarías rehacer la imagen a modo de vector o utilizar una foto de alta resolución y ajustar la resolución del archivo en torno a 300 PPI con el fin de imprimirla sin perder la calidad de la imagen. La densidad de la pantalla de Android funciona de manera similar, a excepción de que no estamos cambiando la resolución del archivo, sino únicamente el tamaño de la imagen. Un buen

estándar a adoptar es 72 PPI.

Ahora supongamos que tomaste un icono en bitmap de 100×100 pixeles de una de tus pantallas de tu diseño base (recuerda que “la base” es un layout de 320×480). Colocando este mismo icono de 100×100 en un dispositivo con una pantalla IDPI haría que la apariencia del icono fuera grande y borrosa. Del mismo modo, colocarlo en otro dispositivo con una pantalla HDPI lo haría verse demasiado pequeño (esto se debe a que el dispositivo tiene más puntos por pulgada que la pantalla MDPI).

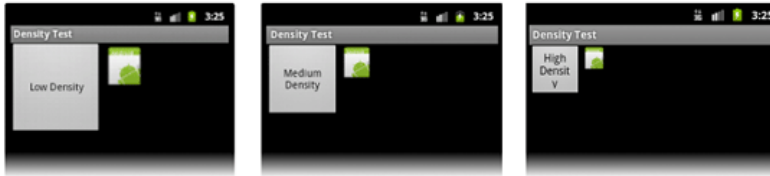


Imagen tomada del artículo original en la que se muestra una aplicación que no tiene soporte a la densidad.

Para ajustar las imágenes a las diferentes densidades de pantalla de los dispositivos, necesitamos seguir una relación de escala de 3:4:6:8 entre los cuatro tamaños de densidad. Usando esto y un poco de matemáticas simples, podremos crear cuatro versiones diferentes de nuestro bitmap para poder trabajar en la aplicación:

- 75×75 para pantallas de baja densidad (x0.75);
- 100×100 para pantallas de media densidad (nuestra base);
- 150×150 para pantallas de alta densidad (x1.5);
- 200×200 para pantallas de muy alta densidad (x2.0)



Imagen tomada del artículo original en la que se muestran las cuatro versiones de una imagen optimizadas para las diferentes densidades de las pantallas.

Una vez de que hayas creado todos los gráficos, puedes organizarlos de la siguiente manera:

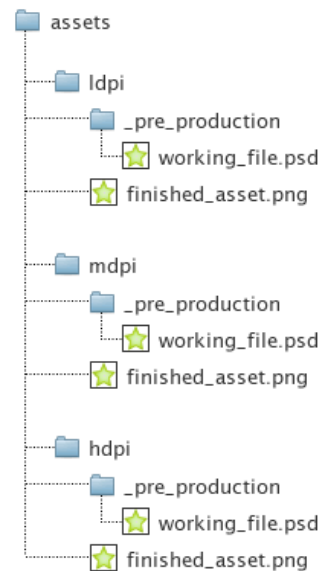


Imagen tomada del artículo original en el que se muestra la organización y nombrado sugerido de la estructura de directorios y archivos. De esta manera podemos utilizar el mismo nombre para cada uno de los archivos de cada densidad de pantalla.

Puede que te confunda un poco el uso de PPI (píxeles por pulgada) para realizar cada versión de las imágenes. Bastará con crear las imágenes bajo un estándar de 72 PPI y escalar las imágenes según sea la versión que necesites crear.

Usando patrones de diseño en Android

A menudo, los clientes preguntan si pueden usar su diseño de aplicación en iPhone para Android. Si lo que estás buscando son atajos, hacer una aplicación para los navegadores web para móviles utilizando una herramienta como [Webkit HML5](#) sea quizás la mejor opción para ti. Sin embargo, para crear aplicaciones nativas en Android, la respuesta es no. ¿Por qué? Porque las convenciones de las interfaces de usuario en Android son diferentes a las del iPhone.

La gran diferencia reside en la tecla "Back", para navegar a las páginas previas. La tecla "Back" en los dispositivos Android es fija y siempre está disponible para el usuario, independientemente de la aplicación. O es una tecla física del dispositivo o se encuentra fija digitalmente en la parte inferior de la pantalla independientemente de cualquier aplicación, tal y como se aprecia en la versión Android 3.x para tablets.

La presencia de una tecla "Back" fuera de la aplicación deja espacio para otros elementos en la parte superior de la pantalla, tales como el logo, título o menú. Mientras que esta convención relacionada con la navegación difiere mucho de la de iOS, existen otras diferencias que en Android se conocen como "patrones de diseño". De acuerdo con Android, un patrón de diseño es una "solución general a un problema recurrente". A continuación se presentan los principales patrones de diseño de Android que se introdujeron desde la versión 2.0.

Dashboard

Este patrón resuelve el problema de tener que navegar entre las diferentes capas de una aplicación. Proporciona una plataforma de lanzamiento alternativa para aplicaciones ricas e interactivas como Facebook, Evernote y LinkedIn.

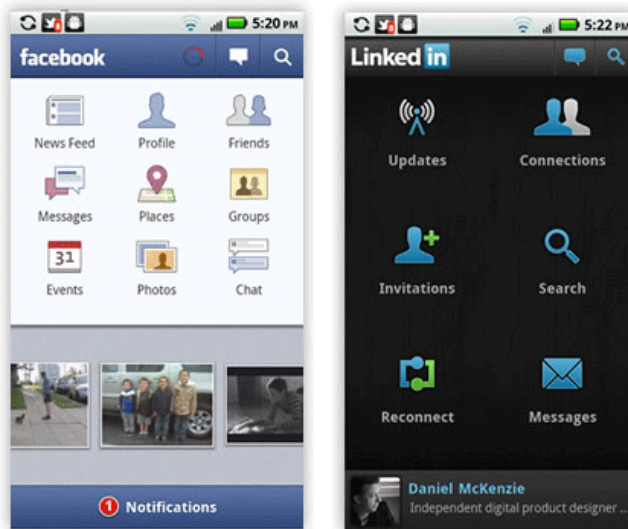


Imagen tomada del artículo original que muestra el patrón de diseño de Dashboard y su uso en las aplicaciones de Facebook y LinekdIn.

Action Bar

La barra de acciones es uno de los patrones de diseño más importantes de Android y que además lo diferencia. Su funcionamiento es muy similar a la de un banner de los sitios web, con el logo o el título generalmente a la izquierda y los elementos de navegación a la derecha. El diseño de la barra de acciones es flexible y permite que se cierren menús y se amplíen las cajas de búsqueda. Generalmente es usada como una característica global en lugar de una característica contextual.



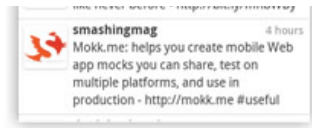


Imagen tomada del artículo original que muestra el patrón de diseño de barra de acciones utilizada por la aplicación de Twitter.

Search Bar

Le brinda al usuario una forma sencilla de buscar por categoría y ofrece además sugerencias de búsqueda.

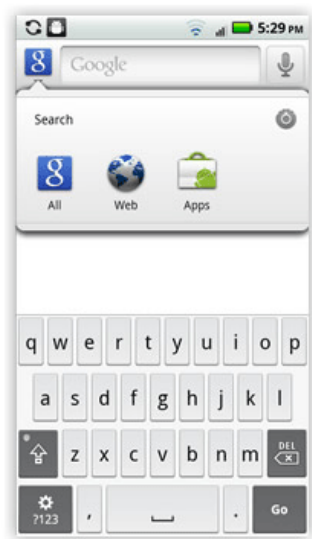


Imagen tomada del artículo original que muestra el patrón de diseño de barra de búsqueda en la aplicación de Google Search.

Quick Action

Este patrón de diseño es similar al comportamiento de los pop-ups en iOS, que le da al usuario acciones contextuales adicionales. Por ejemplo, hacer el gesto de tapping sobre una foto en una aplicación podría lanzar una barra de acción rápida que le permita al usuario compartir la foto.



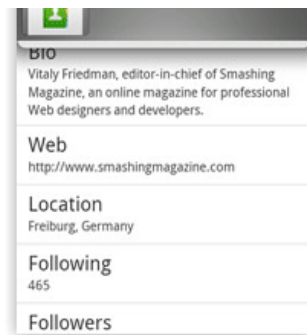


Imagen tomada del artículo original que muestra el patrón de diseño de barra rápida en la aplicación de Twitter.

Widgets

Los widgets le permiten a una aplicación mostrar notificaciones en la pantalla de inicio del usuario. A diferencia de las notificaciones push en iOS que se comportan como modal dialog temporales, los widgets permanecen en la pantalla de inicio.

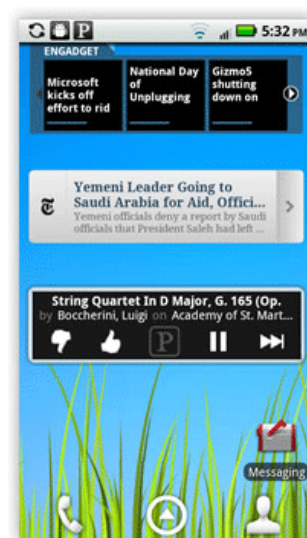


Imagen tomada del artículo original que muestra unos widgets de ejemplo en la pantalla principal del dispositivo.

Utilizar los patrones de diseño establecidos es importante para mantener la experiencia intuitiva y familiar de los usuarios. Hay que tener presente que los usuarios no quieren la misma experiencia del iPhone en su dispositivo Android. Es por ello que la comprensión de los patrones de diseño es el primer paso para aprender a hablar en Android y diseñar una experiencia óptima para los usuarios. ¡Los desarrolladores también te lo agradecerán!

Acerca de las tablets con Android

En la celebración del CESS 2011, las compañías presentaron una gran variedad de tablets con Android, con diferentes tamaños de pantalla. Sin embargo, después de una rápida revisión de las más populares, podemos concluir que los dos tamaños de pantalla más importantes se centran en las medidas de 1280×800 y 800×480 píxeles físicos.

Con el lanzamiento de Android 3 Honeycomb, Google proporciona a los fabricantes de dispositivos una interfaz gráfica especialmente diseñada para tablets, dejando atrás el botón "Back" que ha sido reemplazado por un ancla de navegación y una barra de status del sistema localizada en la parte inferior de la pantalla.

Android 3.0 tiene un aspecto visual renovado, sin dejar de incorporar todos los patrones de diseño que se introdujeron desde la versión 2.0. Una de las grandes diferencias de Honeycomb es que la barra de acciones ha sido actualizada para incluir pestañas y menús desplegables.

Otra nueva característica añadida a Android 3.x es el mecanismo llamado "fragmentos". Un fragmento es un componente auto-contenido en un layout que puede cambiar de tamaño y posición dependiendo de la orientación y el tamaño de la pantalla. Esto además aborda el problema de hacer diseños para diferentes tipos de pantalla dándole a los diseñadores y desarrolladores una manera de hacer sus componentes flexibles dependiendo de las limitantes de pantalla con las que se tope la aplicación. Así, los componentes de la pantalla pueden estirarse, apilarse, expandirse y contraerse, así como mostrarse y ocultarse.

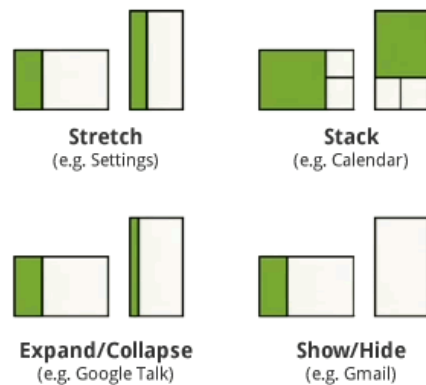


Imagen tomada del artículo original que muestra ejemplos de cómo pueden ser usados los fragmentos.

La próxima versión de Android, llamada Ice cream sándwich, promete traer esta funcionalidad a los smartphones con Android, dándole a los diseñadores y desarrolladores la opción de construir un único layout que se ajuste a las necesidades de la aplicación. Esto podría ser un cambio de paradigma para los diseñadores y

desarrolladores, que tendrán que aprender a pensar en el diseño como si fuese las piezas de un rompecabezas que puede ser ampliado o reducido para ajustarse a la pantalla de los dispositivos. En resumen, esto le permitirá al sistema operativo Android correr en cualquier dispositivo (¡con posibilidades infinitas!).

Un consejo

Trata de comprarte un teléfono o Tablet con Android y tómate el tiempo de descargar aplicaciones para explorar sus interfaces. Con el fin de diseñar para Android, tienes que sumergirte en el entorno y conocerlo a profundidad. Esto puede sonar obvio, pero es siempre sorprendente escuchar que incluso el jefe del producto no tiene un dispositivo con Android.

Conclusiones

Los invito a visitar el apartado de recursos que nos comparte el autor del [artículo original en inglés](#). En lo personal aprendí mucho de esta publicación porque pone sobre la mesa puntos muy importantes del desarrollo de aplicaciones móviles y que a veces se nos olvida tomar en cuenta, espero que te haya gustado la interpretación del mismo.

Publicado por [Lic. Mariano Salvetti](#) en 01:00

+5 [Recomendar esto en Google](#)

Etiquetas: [buenas practicas y patrones](#), [Diseñar Apps Android](#), [Mariano Salvetti](#)

Sin comentarios:

Publicar un comentario en la entrada

Introduce tu comentario...

Comentar como: Seleccionar perfil...

Publicar

Vista previa

[Entrada más reciente](#)

[Página principal](#)

[Entrada antigua](#)

Suscribirse a: [Enviar comentarios \(Atom\)](#)

BLOGS DE REFERENCIA

[JEE Struts2 EJB3 GlassFish v3](#)

ETIQUETAS

[Mariano Salvetti](#) (139) [Android](#) [4.0](#) [Ice Cream Sandwich](#) (38) [Curso Android Rosario](#) (30) [Android Market](#)

(27) [Aprender a programar Android](#) (24) [Android](#) (21) [BootCamp Rosario](#) (19) [Capacitacion Android](#) (17) [Android 3.0](#) (11) [Celulares Android Rosario](#) (11) [Aplicaciones para SmartPhones](#) (10) [curso de Programacion Android en Rosario](#) (7) [Aplicaciones web para celulares](#) (6) [19 de Octubre 2011](#) (5) [Android Facil](#) (5) [Condesa Androideity](#) (5) [Curso de Android en Rosario](#) (5) [Honeycomb](#) (5) [Revista PixelsCode](#) (5) [Activity](#) (4) [AppEngine](#) (4) [Google](#) (4) [Moneda Virtual](#) (4) [novedades Android](#) (4) [versiones de Android](#) (4) [Android 5.0 Jelly Bean](#) (3) [Android Design](#) (3) [Android con buen diseño](#) (3) [Android en Apple](#) (3) [Angry Birds](#) (3) [Aplicacion Web para Android](#) (3) [BlackBerry PlayBook](#) (3) [Descargar Aplicaciones Android Gratis](#) (3) [Eclipse Android actualizar](#) (3) [Getjar](#) (3) [Gingerbread](#) (3) [Google Play](#) (3) [Lanzamiento 2012 GTUG](#) (3) [Nexus S](#) (3) [activaciones Android](#) (3) [Amazon AppStore for Android](#) (2) [AndEngine](#) (2) [Android 2.3 Games](#) (2) [Android Lint](#) (2) [Android Videojuegos](#) (2) [Angry Birds Play](#) (2) [Angry Birds Space](#) (2) [Aplicacion Android](#) (2) [Aplicacion GTUG Argentina](#) (2) [Chromezone Chrome](#) (2) [Ciclo de vida](#) (2) [Comenzar con PhoneGap](#) (2) [Crear una AVD en Eclipse](#) (2) [DattaTec](#) (2) [Desarrollar Android Apps](#) (2) [Evolucion Historica de Android](#) (2) [Fragmentacion de Android](#) (2) [Galaxy Nexus](#) (2) [Grafico Android Vs. IOS](#) (2) [Hello World SDK \(r14\)](#) (2) [Hola Mundo Android 4](#) (2) [Intents](#) (2) [Introduccion a Google Play](#) (2) [JellyBean](#) (2) [Juegos HD para Android](#) (2) [Kernes de Linux en Android](#) (2) [Lifecicle Android apps](#) (2) [Maven2 Eclipse](#) (2) [Otros Markets para Android](#) (2) [Recursos de aplicaciones](#) (2) [Sony](#) (2) [Threads en Android](#) (2) [Windows Mobile](#) (2) [actualizaciones android](#) (2) [git](#) (2) [install Android 4](#) (2) [libgdx](#) (2) [phonegap for Android](#) (2) [publicar en el Android Market](#) (2) [que es android?](#) (2) [versionado de codigo Android](#) (2) [15 juegos para android](#) (1) [3G](#) (1) [4G](#) (1) [4Android.com](#) (1) [ADT 17.0.0](#) (1) [ADT Revision 17](#) (1) [ANDROID vs IPHONE](#) (1) [Actividades](#) (1) [Actividades parámetros entre ellas](#) (1) [Andgry Birds](#) (1) [Androi vs Blackberry Playbook](#) (1) [Android 1 vs Android 2](#) (1) [Android 2012 Rosario](#) (1) [Android 3.0 SDK](#) (1) [Android 4.0.3](#) (1) [Android 4.0.5](#) (1) [Android 4.1](#) (1) [Android ActionBar Sherlock](#) (1) [Android Facebook](#) (1) [Android Git](#) (1) [Android Hola Mundo](#) (1) [Android Market y sus cambios](#) (1) [Android Market y sus millones de descargas](#) (1) [Android Porno](#) (1) [Android Store en China](#) (1) [Android Tools rev 16](#) (1) [Android World Mobile Congress 2012](#) (1) [Android activaciones febrero 2012](#) (1) [Android deteccion de caras](#) (1) [Android en Facebook](#) (1) [Android games](#) (1) [Android o iPhone](#) (1) [Android y la bateria](#) (1) [Androidland en el World Mobile Congress 2012](#) (1) [App Galaxy](#) (1) [App Inventor](#) (1) [AppBrain](#) (1) [AsyncTask](#) (1) [Broadcast receivers](#) (1) [Busqueda por voz](#) (1) [Callbacks](#) (1) [Caracteristicas Android 2.3](#) (1) [Casino Virtual](#) (1) [Chengdu](#) (1) [Chrome for Android](#) (1) [Componentes Android](#) (1) [Content Providers](#) (1) [Cupcake](#) (1) [Cómo usar mejor Android Market](#) (1) [DDMS](#) (1) [Descargas enero 2012](#) (1) [Development VideoGames](#) (1) [Dia de los enamorados](#) (1) [Diferenciacion de Android](#) (1) [Diseñar Apps Android](#) (1) [Diseñar para ICS 4.0](#) (1) [Donut](#) (1) [Dropbox vs Sugarsync](#) (1) [Eclair](#) (1) [Eclipse Error Android](#) (1) [Entorno de desarrollo Android Mac](#) (1) [Entorno de desarrollo Android Windows](#) (1) [Eric Schmidt](#) (1) [FaceDetector Example](#) (1) [FaceDetector Tutorial Android](#) (1) [Facebook para Android](#) (1) [Fin de año 2011 Android](#) (1) [Fotos World Mobile Congress 2012](#) (1) [Froyo](#) (1) [GLOSARIO DE TERMINOS](#) (1) [GTUG Argentina](#) (1) [Game Engine for Android](#) (1) [Google App Engine](#) (1) [Google App Engine Rosario](#) (1) [Google BxV](#) (1) [Google Drive Android](#) (1) [Google Music](#) (1) [Google Play Entretenimiento](#) (1) [Google Play Latinoamerica](#) (1) [Google TV](#) (1) [Google Voice Search](#) (1) [Googlw Bouncer](#) (1) [Google Play movies](#) (1) [Gratis Android](#) (1) [GreenDroid Tutorial](#) (1) [HTC](#) (1) [HTC Nexus One con Juegos HD](#) (1) [HolaMundo PhoneGap](#) (1) [iPhone 4 vs HTC Evo](#) (1) [iPhone vs Android](#) (1) [Ice Cream Sandwich](#) (1) [Infografía](#) (1) [Integracion social](#) (1) [Itunes](#) (1) [Java Life](#) (1) [Java One](#) (1) [Java One 2011 - Java Life](#) (1) [Javier Cano](#) (1) [Larry Page Google](#) (1) [Las 10 apps mas usadas en Android](#) (1) [Lector de Ebook](#) (1) [Librería ActionBar Sherlock](#) (1) [MOTODEV App Summit](#) (1) [Mac OS X Leopard](#) (1) [Market nuevo Play](#) (1) [Market open-source](#) (1) [Maven y Android](#) (1) [Menus en Android](#) (1) [Mikandi](#) (1) [N64oid](#) (1) [Nacional B Argentina](#) (1) [Nexus One](#) (1) [Ninja's Unboxing: History](#) (1) [Nintendo 64 para Android](#) (1) [Nokia Belle](#) (1) [Nombres Android](#) (1) [OAuth](#) (1) [Open Handset Alliance explicacion](#) (1) [Persistencia en Android](#) (1) [PhoneGap en Eclipse](#) (1) [PlayStation](#) (1) [Pornografía en Android](#) (1) [Procesos e Hilos en Android](#) (1) [Programar Android 4](#) (1) [Proxy para Google music](#) (1) [Publicar en Market](#) (1) [R17 Android](#) (1) [Recursos – Introducción](#) (1) [Resolucion de pantalla en Android](#) (1) [Ruleta](#) (1) [SDK Tools](#) (1) [SDK Tools Update Abril 2012](#) (1) [SDK Tools Update Marzo 2012](#) (1) [SDK de Ice Cream Sandwich](#) (1) [Saludo fin de año 2011 Android](#) (1) [Service Ciclo de Vida](#) (1) [Slide me](#) (1) [Slots Royale](#) (1) [Stephen Wozniak Android](#) (1) [Symbian Anna](#) (1) [Tesis de Licenciatura Matematica](#) (1) [Testing App Android](#) (1) [Tragamonedas](#) (1) [Uso De Redes Sociales: Android Vs. IOS](#) (1) [Vanesa Vanteenkiste](#) (1) [Widgets Capa de Aplicaciones](#) (1) [Xoom](#) (1) [Xperia 2011](#) (1) [YAAM](#) (1) [Year of the Dragon](#) (1) [acelerometro](#) (1) [android y Kinect](#) (1) [bluetooth](#) (1) [buenas practicas y patrones](#) (1) [comenzando con libgdx](#) (1) [componentes gráficos para Android](#) (1) [creando JSP](#) (1) [cuidar la bateria del smartphone](#) (1) [disculpas](#) (1) [documentación de Android](#) (1) [entretenimiento digital](#) (1) [estadísticas del Android Market](#) (1) [eventos tecnologicos 2011](#) (1) [features Android](#) (1) [google compra android](#) (1) [gtug Rosario](#) (1) [http://androideity.com/](#) (1) [http://blog.koalite.com/](#) (1) [http://www.androidizados.com](#) (1) [instalar aplicaciones sin android market](#) (1) [juegos en Android](#) (1) [openGL Android](#) (1) [r18](#) (1) [redes wifi](#) (1) [resolucion de pantalla y densidad en Android](#) (1) [servicio propio en](#)

[Android \(1\) sgoliver.net \(1\) tablet \(1\) tipos de procesos en Android \(1\) tutorial cordova](#)
[Android \(1\) versión de Flash en un dispositivo Android \(1\) vogella.com \(1\)](#)

Plantilla Simple. Con la tecnología de [Blogger](#).