

I.E.S. RIBERA
DE CASTILLA

2º DAW - 2021/2022

DISEÑO DE INTERFACES WEB

PRESENTADO POR

Jorge Hernández Flores.

INDICE DEL DOCUMENTO

<i>¿QUÉ SON LOS ÍCONOS DE MATERIALES?</i>	3
---	---

DESARROLLOS	3
--------------------------	---

OBTENER ICONOS	4
-----------------------------	---

LICENCIA	4
EXPLORAR Y DESCARGAR ICONOS INDIVIDUALES.....	4
DESCARGANDO TODO	4
REPOSITORIO DE GIT	4
INSTALACIÓN DE ICONOS DE BOWER.....	5
INSTALACIÓN DE ICONOS DE NPM	5

FUENTE DE ICONOS PARA LA WEB	5
---	---

MÉTODO DE CONFIGURACIÓN 1. USO A TRAVÉS DE FUENTES WEB DE GOOGLE	6
MÉTODO DE CONFIGURACIÓN 2. AUTOHOSPEDAJE	7
USANDO LOS ICONOS EN HTML	10

ESTILO DE ICONOS EN MATERIAL DESIGN	12
DIMENSIONAMIENTO	13
COLORANTE	14

IMÁGENES DE ICONOS PARA LA WEB..... 16

SVG	16
PNG	17

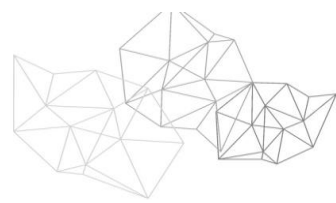
ICONOS PARA ANDROID 18

ICONOS PARA IOS 19

ICONOS EN RTL..... 21

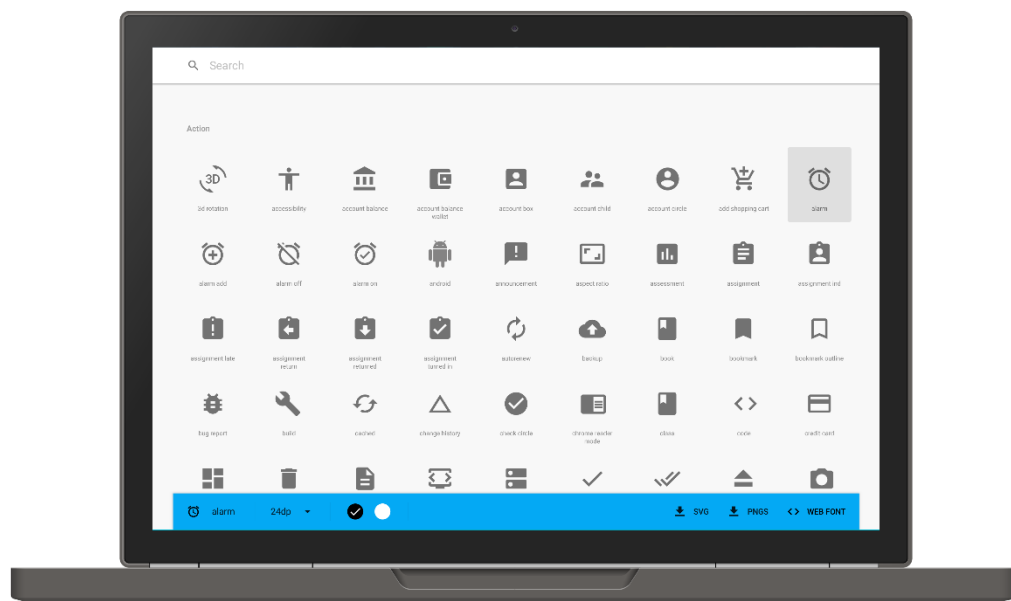
ICONOS RTL EN ANDROID	21
ICONOS RTL EN IOS	24
ICONOS RTL EN LA WEB	26
GENERE SUS PROPIOS ICONOS RTL USANDO IMAGEMAGICK ...	32
¿QUÉ ICONOS DEBEN REFLEJARSE PARA RTL?	32

CONCLUSION 32



DESARROLLO

Vea el conjunto completo de iconos de diseño de materiales en la [Biblioteca de iconos de materiales](#) .



Obtener iconos

Los iconos están disponibles en varios formatos y son adecuados para diferentes tipos de proyectos y plataformas, para desarrolladores en sus aplicaciones y para diseñadores en sus maquetas o prototipos.

Licencia

Hemos puesto estos íconos a su disposición para que los incorpore a sus productos bajo la [Licencia Apache Versión 2.0](#) . Siéntase libre de mezclar y volver a compartir estos íconos y documentación en sus productos. Nos encantaría atribución en su aplicación es *sobre* la pantalla, pero no es necesario. Lo único que le pedimos es que no revenda los iconos en sí.

Explorar y descargar iconos individuales

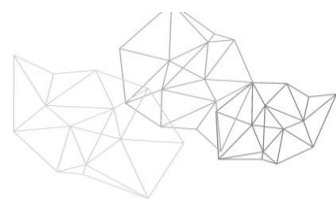
El conjunto completo de iconos de materiales está disponible en la [biblioteca de iconos de materiales](#) . Los iconos están disponibles para descargar en SVG o PNG, formatos adecuados para proyectos web, Android e iOS o para su inclusión en cualquier herramienta de diseño.

Descargando todo

Obtenga el [último archivo zip estable](#) (~ 60 MB) de todos los íconos o la [versión más avanzada del maestro](#).

Repositorio de Git

Los íconos de material están disponibles en el [repositorio de git](#), que contiene el conjunto completo de íconos, incluidos todos los formatos que estamos poniendo a disposición.




```
$ git clone http://github.com/google/material-design-  
icons/
```

Instalación de iconos de bower

Instale los íconos usando el administrador de paquetes de [Bower](#) .

```
$ bower install material-design-icons
```

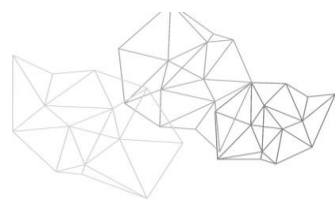
Instalación de iconos de npm

Instale los íconos usando el [administrador de](#) paquetes [npm](#) .

```
$ npm install material-design-icons
```

Fuente de iconos para la web

La fuente de icono de material es la forma más sencilla de incorporar iconos de material en proyectos web. Hemos empaquetado todos los íconos de material en una única fuente que aprovecha las capacidades de representación tipográfica de los navegadores modernos para que los desarrolladores web puedan incorporar fácilmente estos íconos con solo unas pocas líneas de código.



Usar la fuente no solo es el método más conveniente, sino que es eficiente y se ve muy bien:

- Más de 900 iconos, todo desde un único archivo pequeño.
- Se sirve desde los servidores de fuentes web de Google o se puede alojar por sí mismo.
- Compatible con todos los navegadores web modernos.
- Coloreado, dimensionado y posicionado íntegramente con CSS.
- Basado en vectores: se ve muy bien a cualquier escala, pantallas retina, pantallas de visualización de bajo ppp.

La fuente del icono pesa solo 42 KB en su formato woff2 más pequeño y 56 KB en el formato woff estándar. En comparación, los archivos SVG comprimidos con gzip generalmente tendrán un tamaño de alrededor de 62 KB, pero esto se puede reducir considerablemente compilando solo los íconos que necesita en un solo archivo SVG con sprites de símbolos.

Método de configuración 1. Uso a través de fuentes web de Google

La forma más sencilla de configurar fuentes de iconos para su uso en cualquier página web es a través de [Google Web Fonts](https://fonts.google.com/icon) . Todo lo que necesita hacer es incluir una sola línea de HTML:

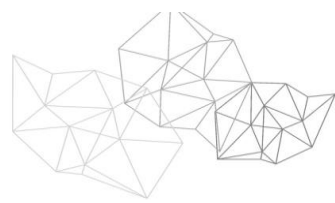
```
<link  
href="https://fonts.googleapis.com/icon?family=Material+Icons"  
  
rel="stylesheet">
```

Al igual que con otras fuentes web de Google, se proporcionará el CSS correcto para activar la fuente "Iconos de material" específica del navegador. Se declarará llamada a una clase CSS adicional `.material-icons`. Cualquier elemento que use esta clase tendrá el CSS correcto para representar estos íconos a partir de la fuente web.

Método de configuración 2. Autohospedaje

Para aquellos que buscan autohospedar la fuente web, es necesaria alguna configuración adicional. Aloje la [fuente](#) del [icono](#) en una ubicación, por ejemplo, `https://example.com/material-icons.woff` agregue la siguiente regla CSS:

```
@font-face {  
  
    font-family: 'Material Icons';  
  
    font-style: normal;  
  
    font-weight: 400;  
  
    src: url(https://example.com/MaterialIcons-Regular.eot);  
    /* For IE6-8 */  
  
    src: local('Material Icons'),  
  
        local('MaterialIcons-Regular'),  
  
        url(https://example.com/MaterialIcons-Regular.woff2)  
    format('woff2'),
```




```
url(https://example.com/MaterialIcons-Regular.woff)
format('woff'),

url(https://example.com/MaterialIcons-Regular.ttf)
format('truetype');

}
```

Además, las reglas CSS para representar el icono deberán declararse para representar la fuente correctamente. Estas reglas se incluyen normalmente como parte de la hoja de estilo de fuentes web de Google, pero deberán incluirse manualmente en sus proyectos cuando se aloje por sí mismo la fuente:

```
.material-icons {

    font-family: 'Material Icons';

    font-weight: normal;

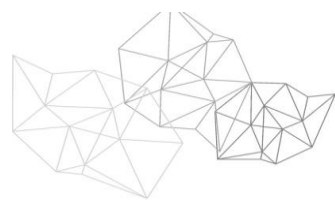
    font-style: normal;

    font-size: 24px; /* Preferred icon size */

    display: inline-block;

    line-height: 1;

    text-transform: none;
```



```
letter-spacing: normal;

word-wrap: normal;

white-space: nowrap;

direction: ltr;


/* Support for all WebKit browsers. */

-webkit-font-smoothing: antialiased;

/* Support for Safari and Chrome. */

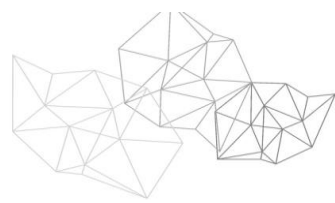
text-rendering: optimizeLegibility;


/* Support for Firefox. */

-moz-osx-font-smoothing: grayscale;


/* Support for IE. */

font-feature-settings: 'liga';
```



```
}
```

Usando los iconos en HTML

Es fácil incorporar íconos en su página web. He aquí un pequeño ejemplo:

cara

```
<span class="material-icons">face</span>
```

Este ejemplo usa una característica tipográfica llamada [ligaduras](#) , que permite la representación de un glifo de icono simplemente usando su nombre textual. El navegador web realiza la sustitución automáticamente y proporciona un código más legible que la referencia de carácter numérico equivalente.

Esta función es compatible con la mayoría de los navegadores modernos en dispositivos móviles y de escritorio.

Navegador	Versión que admite ligaduras
Google Chrome	11
Mozilla Firefox	3,5

Safari de manzana	5
Microsoft IE	10
Microsoft Edge	18
Ópera	15
Apple MobileSafari	iOS 4.2
Navegador de Android	3,0

Para los navegadores que no admiten ligaduras, vuelva a especificar los iconos utilizando referencias de caracteres numéricos como en el ejemplo siguiente:



```
<span class="material-icons">&#xE87C;</span>
```

Busque tanto los nombres de los iconos como los puntos de código en la [biblioteca de iconos de materiales](#) seleccionando cualquier icono y abriendo el panel de fuentes de

iconos. Cada fuente de icono tiene un índice de puntos de código en nuestro repositorio de git que muestra el conjunto completo de nombres y códigos de caracteres ([aquí](#)).

Estilo de iconos en material design

Estos íconos fueron diseñados para seguir las [pautas de diseño de materiales](#) y se ven mejor cuando se usan los tamaños y colores de íconos recomendados. Los estilos a continuación facilitan la aplicación de nuestros tamaños, colores y estados de actividad recomendados.

```
/* Rules for sizing the icon. */

.material-icons.md-18 { font-size: 18px; }

.material-icons.md-24 { font-size: 24px; }

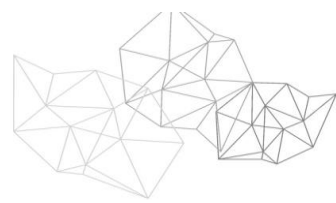
.material-icons.md-36 { font-size: 36px; }

.material-icons.md-48 { font-size: 48px; }


/* Rules for using icons as black on a light background.
*/

.material-icons.md-dark { color: rgba(0, 0, 0, 0.54); }

.material-icons.md-dark.md-inactive { color: rgba(0, 0, 0, 0.26); }
```



```
/* Rules for using icons as white on a dark background. */  
  
.material-icons.md-light { color: rgba(255, 255, 255,  
1); }  
  
.material-icons.md-light.md-inactive { color: rgba(255,  
255, 255, 0.3); }
```

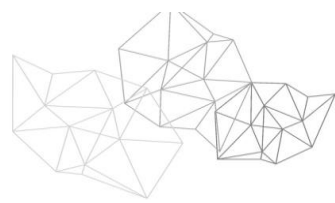
Dimensionamiento

Aunque los íconos de la fuente se pueden escalar a cualquier tamaño, de acuerdo con [las pautas de íconos de material design](#) , recomendamos que se muestren en 18, 24, 36 o 48 px. El valor predeterminado es 24px.

Reglas CSS para las pautas de tamaño estándar de diseño de materiales:

```
.material-icons.md-18 { font-size: 18px; }  
  
.material-icons.md-24 { font-size: 24px; }  
  
.material-icons.md-36 { font-size: 36px; }  
  
.material-icons.md-48 { font-size: 48px; }
```

Los íconos de material se ven mejor a 24px, pero si un ícono debe mostrarse en un tamaño alternativo, el uso de las reglas CSS anteriores puede ayudar:



cara

18px

```
<span class="material-icons md-18">face</span>
```

cara

24px

```
<span class="material-icons md-24">face</span>
```

cara

36px

```
<span class="material-icons md-36">face</span>
```

cara

48px

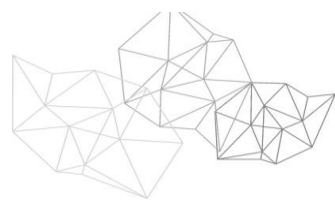
```
<span class="material-icons md-48">face</span>
```

Colorante

El uso de la fuente del icono permite diseñar fácilmente un icono en cualquier color. De acuerdo con [las pautas](#) para los íconos de [material design](#) , para los íconos activos recomendamos usar negro al 54% de opacidad o blanco al 100% de opacidad cuando se muestran sobre fondos claros u oscuros, respectivamente. Si un icono está desactivado o inactivo, utilice negro al 26% o blanco al 30% para fondos claros y oscuros, respectivamente.

A continuación, se muestran algunos ejemplos, utilizando los estilos CSS de material descritos anteriormente:

```
.material-icons.md-dark { color: rgba(0, 0, 0, 0.54); }
```



```
.material-icons.md-dark.md-inactive { color: rgba(0, 0, 0, 0.26); }
```

```
.material-icons.md-light { color: rgba(255, 255, 255, 1); }
```

```
.material-icons.md-light.md-inactive { color: rgba(255, 255, 255, 0.3); }
```

Ejemplo para dibujar un icono sobre un fondo claro con un color de primer plano oscuro:

cara

Normal

```
<span class="material-icons md-dark">face</span>
```

cara

Discapacitado

```
<span class="material-icons md-dark md-inactive">face</span>
```

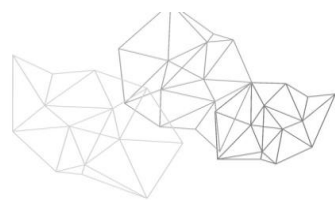
Ejemplo para dibujar un icono sobre un fondo oscuro con un color de primer plano claro:

cara

Normal

```
<span class="material-icons md-light">face</span>
```

cara



Discapacitado

```
<span class="material-icons md-light md-inactive">face</span>
```

Para establecer un color de icono personalizado, defina una regla CSS que especifique el color deseado para la fuente:

```
.material-icons.orange600 { color: #FB8C00; }
```

y luego use la clase cuando se refiera al ícono:

cara

Normal

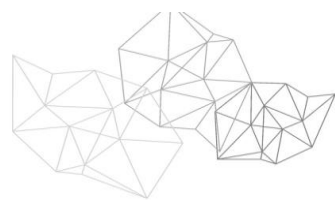
```
<span class="material-icons orange600">face</span>
```

Imágenes de iconos para la web

Los iconos de material también están disponibles como imágenes normales, tanto en formato PNG como SVG.

SVG

Los iconos de material se proporcionan como SVG que son adecuados para proyectos web. Los iconos individuales se pueden descargar de la [biblioteca de iconos de materiales](#) . Los SVG también están disponibles en el [repositorio de git de](#) íconos de material design debajo de la ruta:



```
material-design-icons/*/svg/production/
```

Por ejemplo, los iconos de los mapas están en [maps / svg / production](#) :

```
material-design-icons/maps/svg/production/
```

Si se utilizan varios iconos en un sitio web, se recomienda crear hojas de sprites a partir de las imágenes. Para obtener más información, consulte la documentación en el [directorio de sprites del repositorio de git](#) .

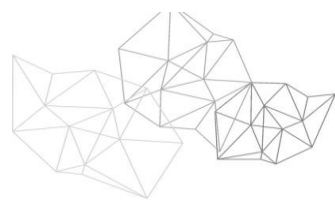
PNG

PNG es la forma más tradicional de mostrar iconos en la web. Nuestras descargas de la [biblioteca de iconos de materiales](#) proporcionan densidades simples y dobles para cada icono. Se denominan 1x_web y 2x_web respectivamente en la descarga. Los iconos también están disponibles en el repositorio de git en:

```
material-design-icons/*/1x_web/
```

```
material-design-icons/*/2x_web/
```

Si se utilizan varios iconos en un sitio web, se recomienda crear hojas de sprites a partir de las imágenes. Para obtener más información, consulte las recomendaciones en el [directorio de sprites en el repositorio de git](#) .



Iconos para Android

Los archivos PNG adecuados para Android están disponibles en la [biblioteca de iconos de materiales](#) . Estos vienen en todas las densidades de pantalla admitidas, por lo que deberían verse bien en cualquier dispositivo.

Los íconos también están disponibles en el [repositorio de git de íconos de material design](#) en la misma combinación de colores y tamaños nombrados de la siguiente manera:

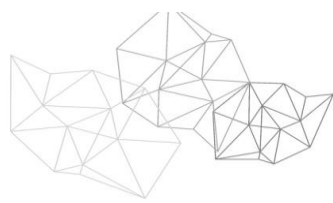
```
*/drawable-  
{{density}}/ic_{{name}}_{{color}}_{{size}}dp.png
```

Se proporciona un VectorDrawable independiente de la densidad que es compatible con Android Lollipop y versiones posteriores:

```
*/drawable-anydpi-v21/ic_{{name}}_black_24dp.xml
```

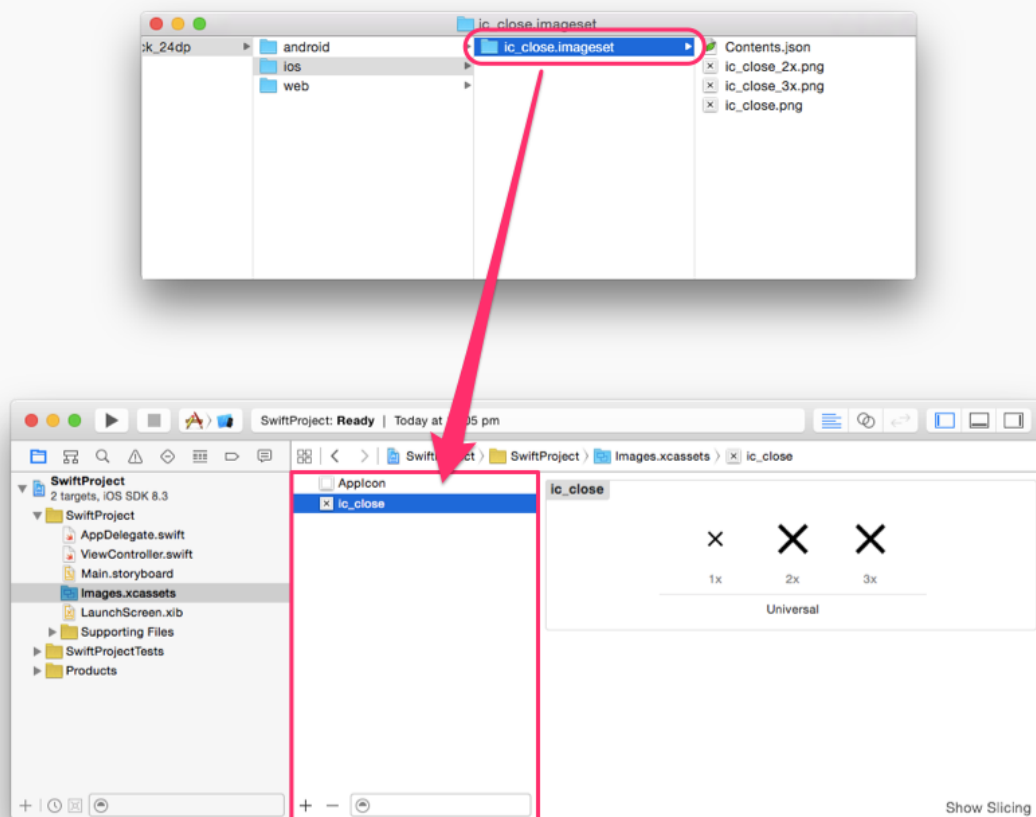
El Vector Drawable actualmente solo está disponible como un ícono negro de 24 dp. Esto es por compatibilidad con nuestro tamaño de icono más estándar. Para renderizar el ícono en un color diferente, use el [tinte dibujable disponible en Android Lollipop](#) .

Al usar Vector Drawable, puede que no sea necesario incluir el PNG de densidad xxxhdpi ya que es poco probable que un dispositivo que admita esa densidad de pantalla no sea compatible con Vector Drawables.



Iconos para iOS

Los íconos de material también funcionan bien dentro de las aplicaciones de iOS. Tanto en la [biblioteca de iconos de materiales](#) como en el [repositorio de git](#), estos iconos están empaquetados en conjuntos de [imágenes de Xcode](#) que funcionarán fácilmente con los [catálogos de activos de Xcode](#) (xcassets). Estos conjuntos de imágenes se pueden agregar a cualquier catálogo de activos de Xcode arrastrándolos a Xcode en el catálogo de activos o copiando la carpeta en la carpeta xcasset.



El conjunto de imágenes contiene imágenes de densidad simple, doble y triple (1x, 2x, 3x) para que funcionen en todas las densidades de pantalla de iOS conocidas. Se proporcionan iconos en blanco y negro, pero recomendamos usar [imageWithRenderingMode de UIImage](#) con [UIImageRenderingModeAlwaysTemplate](#) que permitirá que la imagen se use como una máscara alfa que se puede teñir de cualquier color posible.

Ejemplo de Objective-C:

```
UIButton *button = [[UIButton alloc] init];

UIImage *closeImage =

    [[UIImage imageNamed:@"ic_close"]

    imageWithRenderingMode:UIImageRenderingModeAlwaysTemplate]

;

button.tintColor = [UIColor colorWithWhite:0 alpha:0.54f];

[button setImage:closeImage
forState:UIControlStateNormal];
```

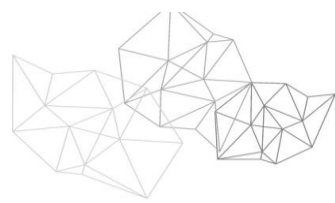
Ejemplo rápido:

```
let button = UIButton()

let closeImage =
UIImage(named:"ic_close")?.imageWithRenderingMode(

    UIImageRenderingMode.AlwaysTemplate)

button.tintColor = UIColor(white:0, alpha:0.54)
```



```
button.setImage(closeImage,  
forState:UIControlState.Normal)
```

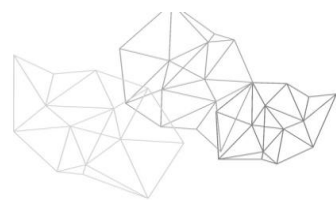
Iconos en RTL

Los idiomas como el árabe y el hebreo se leen de derecha a izquierda (RTL). Para los lenguajes RTL, las IU deben reflejarse para mostrar la mayoría de los elementos en RTL. Cuando una interfaz de usuario está reflejada para RTL, algunos de los iconos también deben reflejarse. Cuando el texto, el diseño y la iconografía se reflejan para admitir interfaces de usuario de derecha a izquierda, todo lo que se relacione con el tiempo debe representarse como un movimiento de derecha a izquierda. Por ejemplo, adelante apunta a la izquierda y atrás apunta a la derecha. Sin embargo, tenga en cuenta que el contexto en el que se coloca el icono también influye en si un icono debe reflejarse o no.

Los iconos solo deben reflejarse si su dirección coincide con otros elementos de la interfaz de usuario en el modo RTL. Cuando un icono representa características visuales de su sitio web que son diferentes en RTL, entonces el icono también debe reflejarse en RTL. Por ejemplo, si los números en una lista numerada están en el lado derecho en el idioma RTL, entonces los números deben estar en el lado derecho del icono reflejado.

Nota: Los iconos que incluyen un signo de interrogación deben reflejarse en árabe y farsi, pero no en hebreo. Para obtener una guía detallada sobre este tema, lea el [artículo de especificaciones de diseño de material bidireccional](#) .

Iconos RTL en Android



[Este artículo para desarrolladores de Android](#) describe en profundidad cómo implementar interfaces de usuario RTL. De forma predeterminada en Android, los iconos no se reflejan cuando se refleja la dirección del diseño. Necesita reflejar específicamente los iconos apropiados cuando sea necesario, ya sea proporcionando activos especializados para lenguajes RTL o utilizando la funcionalidad del marco para reflejar los activos.

Para proporcionar activos especializados para lenguajes RTL, puede utilizar el `ldrtl` calificador en directorios de recursos, como `res/drawable-ldrtl/`. Los recursos dentro de dichos directorios solo se usarán para lenguajes RTL. Para dispositivos que ejecutan Android API 19 o más reciente, el marco también proporciona el [atributo](#) `autoMirrored` para Drawables. Cuando este atributo se establece en verdadero, el elemento de diseño se reflejará automáticamente en los lenguajes RTL.

Usando `autoMirrored`:

```
<vector
xmlns:android="http://schemas.android.com/apk/res/android"

    android:width="24dp"

    android:height="24dp"

    android:viewportWidth="24.0"

    android:viewportHeight="24.0"

    android:tint="?attr/colorControlNormal"

    android:autoMirrored="true">
```

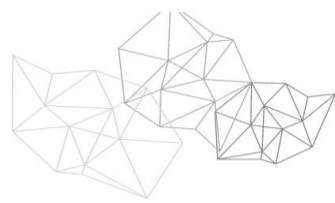
```
<path  
  
    android:fillColor="@android:color/white"  
  
    android:pathData="M20,11H7.83l5.59,-5.59L12,41-8,8  
8,  
  
    8 1.41,-1.41L7.83,13H20v-2z"/>  
  
</vector>
```

Si usar `autoMirrored` o proporcionar recursos dibujables alternativos no es una opción, el atributo `scaleX` de `ImageView` también se puede usar para reflejar elementos dibujables (por ejemplo, proporcionando un diseño específico de RTL en un `res/layout-ldrtl` directorio).

Reflejo dentro del archivo de diseño:

```
<ImageView  
  
    android:id="@+id/my_icon"  
  
    android:layout_width="60dp"  
  
    android:layout_height="60dp"  
  
    android:scaleX="-1" />
```

Por último, los elementos de diseño se pueden duplicar mediante programación.



Verifique manualmente la dirección del diseño usando [getLayoutDirection](#) :

```
if (ViewCompat.getLayoutDirection(view) ==  
ViewCompat.LAYOUT_DIRECTION_RTL) {  
  
    // custom code  
  
}
```

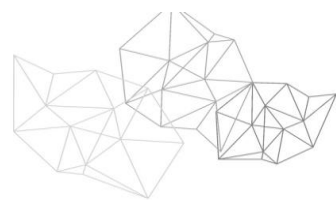
Duplicar el contenido de ImageView mediante programación:

```
imageView.setScaleX(-1);
```

Iconos RTL en iOS

iOS tiene el concepto de un [UISemanticContentAttribute](#) que se adjunta a cada vista. Esto puede ser `unspecified`, `forceLeftToRight`, `forceRightToLeft`, `playbackO spatial`. iOS usa este valor y la configuración (de izquierda a derecha (LTR) / RTL del dispositivo que presenta la interfaz para determinar la dirección de diseño efectiva de la vista. Esta dirección de diseño efectiva determina si se debe reflejar o no una imagen cuando se muestra.

De forma predeterminada, el contenido semántico de las imágenes está configurado en `unspecified`. Esto hace que se reflejen en modo RTL. Si no desea que un icono se refleje nunca, debe configurarlo explícitamente `forceLeftToRight`. Apple señala algunas excepciones que no deben reflejarse, como la reproducción de medios (avance rápido, rebobinado, etc.), notas musicales, imágenes que indican el paso del tiempo, etc.



Ejemplo de Objective-C:

```
// Prevent an icon from being mirrored in RTL mode

UIImage *icon = [UIImage imageNamed:@"my_icon.png"];

UIImageView *iconView = [[UIImageView alloc]
initWithImage:icon];

iconView.semanticContentAttribute =

    UISemanticContentAttributeForceLeftToRight;
```

Ejemplo rápido:

```
// Prevent an icon from being mirrored in RTL mode

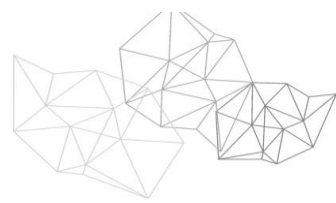
let iconImage = UIImage.init(named: "my_icon.png")

let iconView = UIImageView.init(image: iconImage)

iconView.semanticContentAttribute = .forceLeftToRight;
```

Para obtener documentación más detallada sobre cómo implementar RTL en iOS y macOS, consulte [la documentación de RTL de Apple](#) .

Se agregó contenido semántico en iOS 9. Si admite versiones anteriores de iOS, el [marco de internacionalización de materiales](#) respalda algunas de las funciones a iOS 8.



Iconos RTL en la web

Recomendamos el siguiente artículo como introducción a RTL en la web: <https://hacks.mozilla.org/2015/09/building-rtl-aware-web-apps-and-websites-part-1> <https://hacks.mozilla.org/2015/10/building-rtl-awareness-web-apps-websites-part-2>

De forma predeterminada en la web, los iconos no se reflejan cuando se refleja la dirección del diseño. Necesita reflejar específicamente los **iconos apropiados** cuando sea necesario.

El siguiente ejemplo muestra cómo implementar una regla RTL CSS simple. También puede [verlo en codepen](#).

page.html

```
<html dir="rtl">

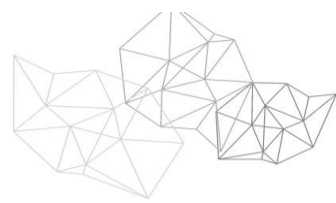
</html>
```

page.css

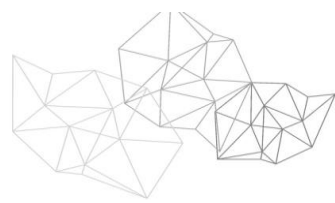
```
html[dir="rtl"] .icon {

  -moz-transform: scaleX(-1);

  -o-transform: scaleX(-1);
```



```
-webkit-transform: scaleX(-1);
```



arrow_backflecha hacia atrás

arrow_back_iosflecha hacia atrás ios

arrow
adelante

arrow_forward_iosflecha hacia
adelante ios

arrow_leftflecha izquierda

arrow

asignaciónasignación

asignación_devolucióndevolución
de asignación

retro

batería_desconocidabatería
desconocida

call_madeLlamada hecha

call_r

call_missedLlamada perdida

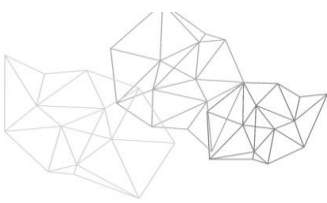
call_missed_outgoingLlamada
saliente perdida

call_r

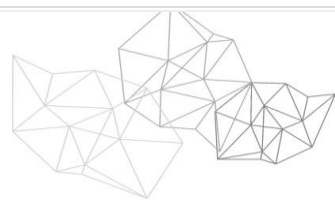
call_splitLlamada dividida

chevron_leftchevron a la izquierda

chevr



chrome_reader_mode modo lector de cromo	device_unknown dispositivo desconocido	dvr dvr
event_note nota de evento	Featured_play_list lista de reproducción destacada	video vídeo
primera página primera página	vuelo_land tierra de vuelo	vuelo vuelo
format_indent_decrease disminución de sangría de formato	format_indent_increase aumento de sangría de formato	formatos formatos
hacia adelante hacia adelante	funciones funciones	ayuda
help_outline esquema de ayuda	aporte aporte	keyboards del teclado



keyboard_tabpestaña del teclado

etiquetaetiqueta

label
importar

label_outlinecontorno de la etiqueta

última páginaúltima página

lanza

lista
lista

ayuda en vivoayuda en vivo

mobi
pantalla

multiline_chartgráfico multilínea

navegar_antesnavegar antes

navig
continua

la próxima semanala próxima semana

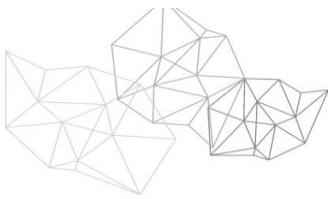
NotaNota

open

playlist_addañadir lista de reproducción

queue_musicCola de musica

reha



respuestarespuesta

responder a todosresponder a todos

comp
pantalla

enviarenviar

Texto cortotexto corto

show

clasificarclasificar

star_halfmitad estrella

tema

trending_flattendencia plana

toctoc

trend

trending_uptendencia al alza

deshacerdeshacer

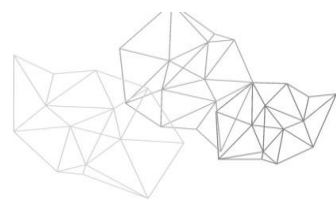
ver li

view_quiltver edredón

wrap_textajustar texto

```
transform: scaleX(-1);
```

```
filter: FlipH;
```




```
-ms-filter: "FlipH";  
  
}
```

Genere sus propios iconos RTL usando ImageMagick

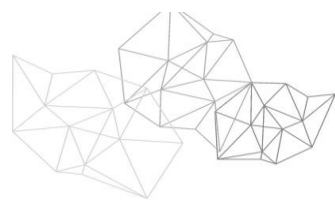
Si reflejar los iconos en el código no es una opción, puede usar [ImageMagick](#) para reflejar la imagen horizontalmente.

```
convert -flop my_icon.png my_icon_rtl.png
```

¿Qué iconos deben reflejarse para RTL?

A continuación, se muestra una lista de iconos que se pueden duplicar mediante programación en RTL:

CONCLUSIÓN



FIN