



EBook Gratis

APRENDIZAJE webview

Free unaffiliated eBook created from
Stack Overflow contributors.

#webview

Tabla de contenido

Acerca de.....	1
Capítulo 1: Empezando con webview.....	2
Observaciones.....	2
Examples.....	2
Instalación o configuración.....	2
Capítulo 2: Android WebView escuchando eventos de la página.....	3
Introducción.....	3
Examples.....	3
Registrar WebViewClient personalizado.....	3
Capítulo 3: WebView móvil.....	5
Introducción.....	5
Examples.....	5
Ejemplo mínimo de Android WebView.....	5
Creditos.....	7

Acerca de

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [webview](#)

It is an unofficial and free webview ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official webview.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Capítulo 1: Empezando con webview

Observaciones

Esta sección proporciona una descripción general de qué es webview y por qué un desarrollador puede querer usarlo.

También debe mencionar cualquier tema importante dentro de webview y vincular a los temas relacionados. Dado que la Documentación para webview es nueva, es posible que deba crear versiones iniciales de los temas relacionados.

Examples

Instalación o configuración

Instrucciones detalladas sobre cómo configurar o instalar webview.

Lea **Empezando con webview en línea**: <https://riptutorial.com/es/webview/topic/9282/empezando-con-webview>

Capítulo 2: Android WebView escuchando eventos de la página

Introducción

WebView componente WebView permite proporcionar un objeto cliente personalizado que amplíe la clase de [WebViewClient](#) para escuchar y controlar el comportamiento de `WebView`.

Examples

Registrar WebViewClient personalizado

```
package com.j2ko.webviewapp;

import android.graphics.Bitmap;
import android.os.Build;
import android.os.Bundle;
import android.support.annotation.RequiresApi;
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
import android.webkit.WebResourceRequest;
import android.webkit.WebView;
import android.webkit.WebViewClient;

public class MainActivity extends AppCompatActivity {

    private static final String TAG = "MainActivity";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        WebView webView = (WebView) findViewById(R.id.webview);
        webView.setWebViewClient(new WebViewClient() {
            @Override
            public void onPageStarted(WebView view, String url, Bitmap favicon) {
                super.onPageStarted(view, url, favicon);
                Log.d(TAG, "onPageStarted: with url " + url);
            }

            @Override
            public void onPageFinished(WebView view, String url) {
                super.onPageFinished(view, url);
                Log.d(TAG, "onPageFinished: for " + url);
            }

            @Override
            public void onPageCommitVisible(WebView view, String url) {
                super.onPageCommitVisible(view, url);
                Log.d(TAG, "onPageCommitVisible: for " + url);
            }
        });
    }
}
```

```

@Override
public boolean shouldOverrideUrlLoading(WebView view, String url) {
    Log.d(TAG, "shouldOverrideUrlLoading: for " + url);
    return super.shouldOverrideUrlLoading(view, url);
}

@RequiresApi(api = Build.VERSION_CODES.LOLLIPOP)
@Override
public boolean shouldOverrideUrlLoading(WebView view, WebResourceRequest request)
{
    Log.d(TAG, "shouldOverrideUrlLoading: for " + request.getUrl());
    return super.shouldOverrideUrlLoading(view, request);
}

@Override
public void onLoadResource(WebView view, String url) {
    super.onLoadResource(view, url);
    Log.d(TAG, "onLoadResource: " + url);
}
});
webView.loadUrl("http://stackoverflow.com/");
}
}

```

Salida:

```

D/MainActivity: onPageStarted: with url http://stackoverflow.com/
D/MainActivity: onLoadResource: http://stackoverflow.com/
D/MainActivity: onLoadResource:
https://cdn.sstatic.net/Sites/stackoverflow/mobile.css?v=88186b623ab1
D/MainActivity: onPageFinished: for http://stackoverflow.com/
D/MainActivity: onLoadResource:
https://cdn.sstatic.net/Sites/stackoverflow/img/favicon.ico?v=4f32ecc8f43d
D/MainActivity: onPageCommitVisible: for http://stackoverflow.com/

```

Lea Android WebView escuchando eventos de la página en línea:

<https://riptutorial.com/es/webview/topic/9284/android-webview-escuchando-eventos-de-la-pagina>

Capítulo 3: WebView móvil

Introducción

`WebView` es un componente de la `UI` que se podría usar para incrustar una página web en su aplicación. Tanto `iOS` como `Android` proporcionan componentes `WebView` como parte de su `SDK`. `WebView` componentes de `WebView` basan principalmente en el proyecto `Chromium` y funcionan con el motor de renderizado `Blink` (anteriormente `WebKit`).

Examples

Ejemplo mínimo de Android WebView

En su diseño de actividad, `activity_main.xml` especifique el componente `WebView` siguiente manera:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <WebView
        android:id="@+id/webview"
        android:layout_width="match_parent"
        android:layout_height="match_parent">
    </WebView>
</RelativeLayout>
```

Luego, en su `MainActivity.java` obtenga el componente `WebView` y cargue alguna URL:

```
package com.j2ko.webviewapp;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.webkit.WebView;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        WebView webView = (WebView) findViewById(R.id.webview);
        webView.loadUrl("http://stackoverflow.com/");
    }
}
```

También debe especificar el permiso de `INTERNET` en su archivo `AndroidManifest.xml`, de lo contrario, `WebView` no podrá cargar la página:

```
<uses-permission android:name="android.permission.INTERNET" />
```

Lea WebView móvil en línea: <https://riptutorial.com/es/webview/topic/9283/webview-movil>

Creditos

S. No	Capítulos	Contributors
1	Empezando con webview	Community
2	Android WebView escuchando eventos de la página	j2ko
3	WebView móvil	j2ko