

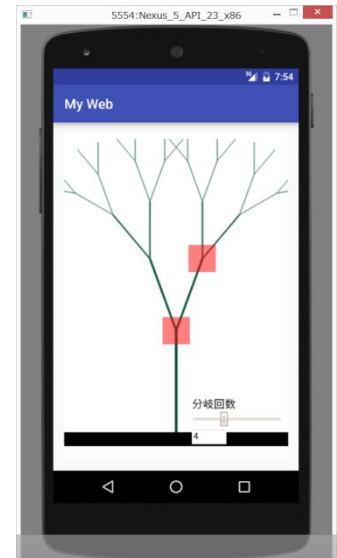
ネイティブアプリのように見えるが、中身は HTML ファイルを Android の WebView で表示しているアプリで、その仕組みからガワアプリ (ガワネイティブ、ハイブリッドアプリ) と呼ばれる。

## 制作手順①②③

### ① Layout の xml ファイルに WebView を配置

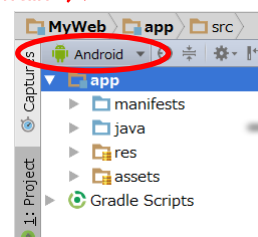
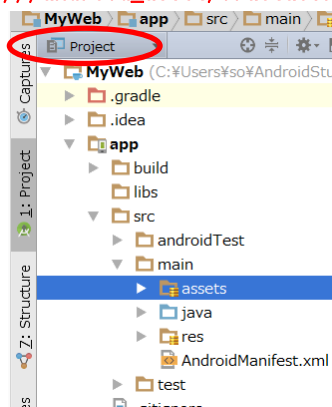
activity\_main.xml

```
<?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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin" tools:context=".MainActivity">
    <WebView
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:id="@+id/webView"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true" />
</RelativeLayout>
```



### ② MainActivity の onCreate メソッドで WebView に loadUrl で Web ページを指定

```
public class MainActivity extends AppCompatActivity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        <b>WebView myWebView = (WebView) findViewById(R.id.webView);</b>
        <b>WebSettings webSettings = myWebView.getSettings();</b>
        <b>webSettings.setJavaScriptEnabled(true);</b>
        <b>myWebView.loadUrl("http://buturi.heteml.jp/js/fractalTree.html");</b>
        //または assets フォルダを src/main/ に作成し、その中のファイル fractalTree.html を読み込む場合は
        //myWebView.loadUrl("file:///android_asset/fractalTree.html");
    }
}
```



③ インターネット接続が必要な場合は、マニフェストで **internet** 接続を許可する。

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" package="com.example.so.myweb" >
<uses-permission android:name="android.permission.INTERNET" />
  <application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:supportsRtl="true"
    android:theme="@style/AppTheme" >
    <activity android:name=".MainActivity" >
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
    </activity>
  </application>
</manifest>
```

参考

- ・ウェブアプリの WebView への組み込み (Android 開発者ガイド > ウェブアプリ > )

<http://www.techdoctranslator.com/android/webapps/webview>

**ガワネイティブ** (<http://www.sophia-it.com/content/ガワネイティブ>)

別名：側ネイティブ、ガワだけネイティブ、ガワだけアプリ、ハイブリッドアプリ

**ガワネイティブ**とは、スマートフォン向けアプリの区分に用いられる用語で、ネイティブアプリとして端末にインストールされたアプリ内で、Web アプリと同様に Web 上のコンテンツを表示する形式のアプリである。

ネイティブアプリは端末にインストールされ、端末内で実行処理される。動作が比較的速く、操作性・機能性に優れるが、プラットフォームごとにアプリを開発する必要があり、開発障壁が比較的高い。他方、Web アプリは、ネットワーク上で処理を行いコンテンツをダウンロードして表示する方式である。Web アプリは主に Web ブラウザを通じて利用される。データや機能がサーバー側にあるためリアルタイムに更新・アップデートすることが可能だが、操作のたびに通信が必要となり、操作が比較的遅いという難点がある。

ガワネイティブは、アプリ自体はネイティブアプリであるが、コンテンツ自体は Web 上から取得して読み込むことで、操作のレスポンスをより早く、開発をより迅速に、コンテンツはより豊富にといった両取りの実現を期待できる方式として、2013 年 7 月現在、注目を集めている。

**WebView**

WebView のレンダリングエンジンは、Android 4.3 までは WebKit, 4.4 以降では Chromiub Blink になった。また、Android4.4 までは、OS に WebView コンポーネントが組み込まれていたが、5.0 からは OS 組み込みではなく、Android System WebView という名称で、Google Play Store からアップデートされるようになった。Google は WebKit ベースの WebView はこれ以上パッチを作成しない。