

Pertemuan 7

Webkit

A. Materi

- Webkit

B. Kompetensi Dasar

- Menerapkan webkit pada aplikasi android

C. Tujuan

- Mahasiswa mampu menerapkan webkit pada aplikasi android

D. Dasar Teori

Pada aplikasi android dapat ditempelkan built in web browser sebagai widget pada activity, untuk menampilkan HTML material atau kemampuan dari browser Android browser berbasis webkit . Android menggunakan widget WebView sebagai host dari halaman browser. Aplikasi yang menggunakan komponen WebView membutuhkan INTERNET permission.

Browser mengakses internet menggunakan koneksi yang ada Webkit digunakan untuk menampilkan web page termasuk method untuk :

- Navigasi forward atau backward melalui history
- Zoom in and out
- Menampilkan pencarian tesk
- Load data
- Stop loading dan sebagainya

Untuk Activity yang mengakses internet dan menampilkan halaman web pada WebView, harus ditambahkan INTERNET permissions pada file Android Manifest :

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

Berikut user permission apabila diterapkan pada file AndroidManifest

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="id.ac.aknpacitan.bab7_webkit" >

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="Bab7_WebKit"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="Bab7_WebKit" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
    <uses-permission android:name="android.permission.INTERNET"/>
</manifest>

```

Tambahkan script pada kotak merah sebagai permission untuk mengakses internet.

Seperti komponen-komponen widget lainnya, untuk menambahkan widget webview dengan cara mendrag komponen WebView ke layout atau menambahkan koding pada file .xml sebagai berikut :

```

<LinearLayout 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">

    <WebView
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:id="@+id/webView"
        android:layout_below="@+id/textView" />

</LinearLayout>

```

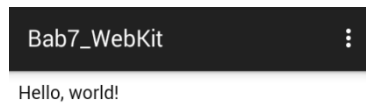
WebView masih berupa komponen widget yang belum di handle penggunaannya. Untuk handle webView deklarasi webView pada class .java kemudian program sesuai kebutuhan. Berikut script pada file .java yang handle file .xml diatas :

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    WebView browser = (WebView) findViewById(R.id.webView);
}

```

Coding diatas masih mendeklarasikan webView. Tambahkan script dibawah ini supaya webView meload HTML sederhana seperti pada **Gambar 7.1**.



Gambar 7.1. Webkit load HTML sederhana.

Webkit juga dapat menampilkan suatu website dengan cara load url web yang dituju. Berikut program untuk menampilkan :

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    WebView browser = (WebView) findViewById(R.id.webView);
    browser.loadUrl("http://aknpacitan.ac.id");
    browser.getSettings().setJavaScriptEnabled(true);
}
```

Script diatas me-load url <http://aknpacitan.ac.id> , ketika di running webkit akan menampilkan seperti **Gambar 7.2.**



Gambar 7.2. WebView load website AKN Pacitan

Posisi dari **Gambar 7.2**. Masih belum terletak di fit dengan layar, tambahkan script berikut untuk menampilkan webView pada **Gambar 7.3**.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    WebView browser =(WebView) findViewById(R.id.webView);
    browser.loadUrl("http://aknpacitan.ac.id");
    browser.getSettings().setJavaScriptEnabled(true);
    browser.setInitialScale(1);
    browser.getSettings().setLoadWithOverviewMode(true);
    browser.getSettings().setUseWideViewPort(true);
}
```



Gambar 7.3 Menampilkan WebView posisi center

WebView juga bisa ditambahkan fasilitas Zoom in dan Zoom out seperti pada **Gambar 7.4**. dengan menambahkan script sebagai berikut :

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    WebView browser =(WebView) findViewById(R.id.webView);
    browser.loadUrl("http://aknpacitan.ac.id");
    browser.getSettings().setJavaScriptEnabled(true);
    browser.setInitialScale(1);
    browser.getSettings().setLoadWithOverviewMode(true);
    browser.getSettings().setUseWideViewPort(true);

    browser.getSettings().setBuiltInZoomControls(true);
    browser.getSettings().setDisplayZoomControls(true);
}
```

Method `setBuiltInZoomControls(true)` dan `setDisplayZoomControl(true)` digunakan untuk menambahkan zoom in dan zoom out



Gambar 7.4. Fasilitas zoom in/out pada WebView

Pada webView ketika menu web diklik website akan dijalankan pada browser default dari device. Supaya website selalu ditampilkan pada webView maka harus ditambahkan coding yang di dalam kotak merah.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    browser = (WebView) findViewById(R.id.webView);
    browser.loadUrl("http://aknpacitan.ac.id");
    browser.getSettings().setJavaScriptEnabled(true);
    browser.setInitialScale(1);
    browser.getSettings().setLoadWithOverviewMode(true);
    browser.getSettings().setUseWideViewPort(true);

    browser.getSettings().setBuiltInZoomControls(true);
    browser.getSettings().setDisplayZoomControls(true);
    browser.setWebViewClient(new WebViewClient());
}
```

Webkit mendukung Navigasi forward atau backward melalui history, Zoom in and out, menampilkan pencarian tesk, load data, stop loading dan sebagainya.

Berikut contoh penambahan menu pada webkit. Tambahkan item menu pada menu_main.xml sebagai berikut :

```

<menu xmlns:android="http://schemas.android.com/apk/res/android"
      xmlns:app="http://schemas.android.com/apk/res-auto"
      xmlns:tools="http://schemas.android.com/tools" tools:context=".MainActivity">
    <item android:id="@+id/action_refresh" android:title="refresh"
          android:orderInCategory="100" app:showAsAction="never" />
    <item android:id="@+id/action_stop" android:title="stop"
          android:orderInCategory="100" app:showAsAction="never" />
    <item android:id="@+id/action_back" android:title="back"
          android:orderInCategory="100" app:showAsAction="never" />
    <item android:id="@+id/action_forward" android:title="forward"
          android:orderInCategory="100" app:showAsAction="never" />
</menu>

```

Untuk memberikan aksi ketika salah satu menu dipilih, tambahkan script berikut pada method `onOptionsItemSelected()` → berada pada class .java

```

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();

    //noinspection SimplifiableIfStatement
    if (id == R.id.action_refresh) {
        browser.reload();
        return true;
    }
    if (id == R.id.action_back) {
        browser.goBack();
        return true;
    }
    if (id == R.id.action_stop) {
        browser.stopLoading();
        return true;
    }
    if (id == R.id.action_forward) {
        browser.goForward();
        return true;
    }
    return super.onOptionsItemSelected(item);
}

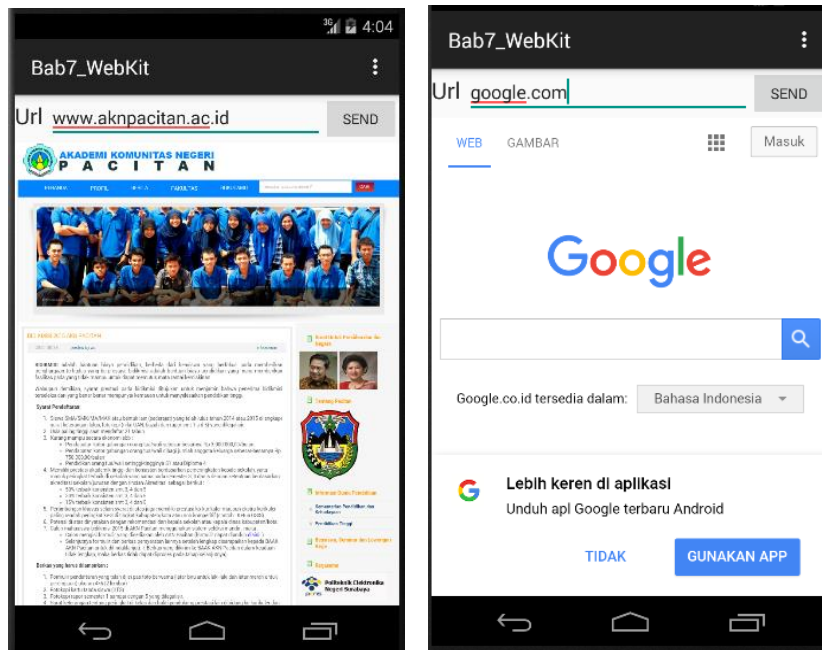
```

E. Percobaan

Lakukan langkah-langkah membuat webView pada dasar teori!

F. Praktikum

1. Buat webView yang me-load url inputan dari user. Seperti **Gambar 7.5**.



Gambar 7.5. Hasil praktikum 7

2. Tambahkan menu : refresh, back, stop dan forward! Atur android:showAsAction=”always”!