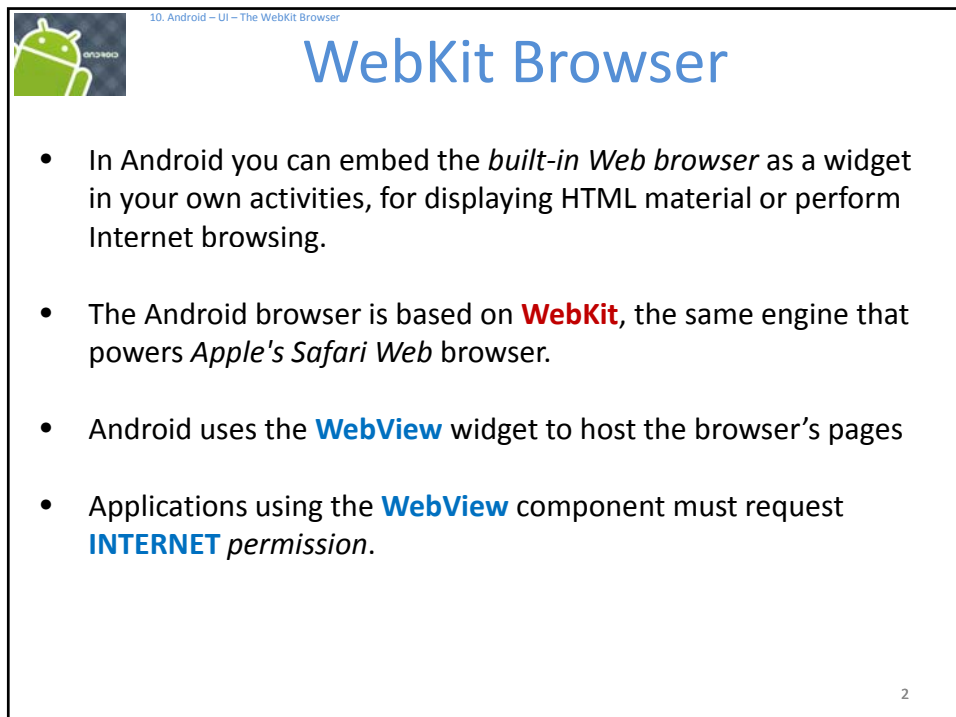*10*

# Android
# The WebKit Browser

Victor Matos
Cleveland State University

Notes are based on:
The Busy Coder's Guide to Android Development
by Mark L. Murphy
Copyright © 2008-2009 CommonsWare, LLC.
ISBN: 978-0-9816780-0-9
&
Android Developers
http://developer.android.com/index.html

---

10. Android – UI – The WebKit Browser

# WebKit Browser

- In Android you can embed the *built-in Web browser* as a widget in your own activities, for displaying HTML material or perform Internet browsing.

- The Android browser is based on **WebKit**, the same engine that powers *Apple's Safari Web* browser.

- Android uses the **WebView** widget to host the browser's pages

- Applications using the **WebView** component must request **INTERNET** *permission*.

2

# WebKit Browser

**Browsing Power**

The browser will access the Internet through whatever means are available to that specific device at the present time (WiFi, cellular network, Bluetooth-tethered phone, etc.).

The **WebKit** rendering engine used to display web pages includes methods to

1. navigate forward and backward through a history,
2. zoom in and out,
3. perform text searches,
4. load data
5. stop loading and
6. more.

3

# WebKit Browser

**Warning**

In order for your Activity to access the Internet and load web pages in a *WebView*, you must add the *INTERNET* permissions to your Android Manifest file:

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

This must be a child of the *<manifest>* element.

*(see next example)*

4

# WebKit Browser

**Example: A simple browsing experience**
Let's go e-shopping

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
    <WebView
        android:id="@+id/webkit"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
    />
</LinearLayout>
```

5

---

# WebKit Browser

**Example: A simple browsing experience**
Let's go e-shopping

```java
package cis493.demoui;

import android.os.Bundle;
import android.app.Activity;
import android.webkit.WebView;

public class AndDemoUI extends Activity {
WebView browser;

    @Override
    public void onCreate(Bundle icicle) {
        super.onCreate(icicle);
        setContentView(R.layout.main);
        browser=(WebView)findViewById(R.id.webkit);
        browser.loadUrl("http://eBay.com");
        browser.getSettings().setJavaScriptEnabled(true);
    }
}
```

This app is hard-wired to eBay

6

3

# WebKit Browser

**Example: A simple browsing experience**

Let's go e-shopping    -    Manifest

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
 package="cis493.demoui" android:versionCode="1" android:versionName="1.0">

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


 <application android:icon="@drawable/icon" android:label="@string/app_name">
    <activity android:name=".AndDemoUI" android:label="@string/app_name">
    <intent-filter>
      <action android:name="android.intent.action.MAIN" />
      <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
    </activity>
 </application>

 <uses-sdk android:minSdkVersion="3" />

</manifest>
```

7

# WebKit Browser

## Warning

If you set the URL to a site whose pages depend on *Javascript* you may see an empty, white screen.

By *default* **Javascript** is turned **off** in WebView widgets.

If you want to enable Javascript, call :

```
 myWebView.setSettings().setJavaScriptEnabled(true);
```

on the WebView instance.

*To be discussed later in this chapter.*

8

4

# WebKit Browser

## Warning

Under SDK 1.6 a WebView has a **built-in Option Menu**



Using **Go** option          Using **More** option          9

# WebKit Browser

## Loading Data  *.loadData(…)*

You may directly provide the HTML to be displayed by the browser
(*a user manual for instance, or the actual app interface created as HTML instead
of using the native Android UI framework*).

```
package cis493.demoui;

import android.os.Bundle;
import android.app.Activity;
import android.webkit.WebView;

public class AndDemoUI extends Activity {
    WebView browser;

    @Override
    public void onCreate(Bundle icicle) {
        super.onCreate(icicle);
        setContentView(R.layout.main);
        browser=(WebView)findViewById(R.id.webkit);

        browser.loadData("<html><body>Hello, world!</body></html>",
                        "text/html",
                        "UTF-8");
    }
}
```

Use same layout and manifest of previous example

10

5

10. Android – UI – The WebKit Browser

# WebKit Browser

**Browser Commands**

There is no navigation toolbar with the WebView widget (*saving space*).
You could supply the UI –such as a Menu– to execute the following operations:

- **reload()** to refresh the currently-viewed Web page
- **goBack()** to go back one step in the browser history, and canGoBack() to determine if there is any history to trace back
- **goForward()** to go forward one step in the browser history, and canGoForward() to determine if there is any history to go forward to
- **goBackOrForward()** to go backwards or forwards in the browser history, where *negative/positive* numbers represent a count of steps to go
- **canGoBackOrForward()** to see if the browser can go backwards or forwards the stated number of steps (following the same positive/negative convention as goBackOrForward())
- **clearCache()** to clear the browser resource cache and **clearHistory()** to clear the browsing history

11

---

10. Android – UI – The WebKit Browser

# WebKit Browser

Using our running example:

```
browser.goBack();
browser.goForward();
browser.goBackOrForward(-2);
browser.goBackOrForward(+2);
browser.canGoBack();
browser.canGoForward();
browser.canGoBackOrForward(-2);
browser.canGoBackOrForward(+2);
browser.clearCache(true);
browser.clearHistory();
browser.stopLoading();
```

12

10. Android – UI – The WebKit Browser

# WebKit Browser

## Questions ?

13