Lesson 1

Android Development Introduction

Victor Matos
Cleveland State University

Portions of this page are reproduced from work created and shared by Google and used according to terms described in the Creative Commons 3.0 Attribution License.

1. Android – How-do-cellular-phones-work?

Mobile Phone Evolution

1876
- ***Alexander Graham Bell*** became the first to receive a patent for the electric phone.

1936
- **Alfred Gross.** Case Tech OH (Case Western Reserve University). Invented/Patented Walkie-talkie, CB radio, Telephone Pager.

Chester Gould

1975
- **Dr. Martin Cooper** invented first commercial portable Motorola radio phone

2007
- iPhone
- Android

Images from: http://en.wikipedia.org/wiki/Dick_Tracy http://en.wikipedia.org/wiki/Martin_Cooper_(inventor)

Hardware: What is inside a Cellular Phone?

Oversimplifying…

Cellular phone = radio + computer*

Industries ⟵ ∑ Software + Telecom+ Semiconductor + Marketing

Hardware: Reusing Cell Phone Frequencies

Great concept !!!

The core idea behind cellular phones is the division of a large city into small areas called **cells**.

Each hexagonal cell covers approx. 10 sq miles (26 km^2)

Base stations use low-power transmitters, therefore the same frequencies can be reused in non-contiguous cells.

Software: What is Android?

- Android is a Linux-based operating system for mobile devices.
- The system is being developed by the Open Handset Alliance and Google Inc.
- Android is an open-source project and is distributed free of charge.
- The operating system has a number of native applications supporting telephony, messaging, emailing, contact management, calendar, entertainment, multimedia experience, location services, mapping, social interaction, etc.
- Third party Java developers can use the Android API to extend the functionality of the devices.
- Google provides a on-line electronic market for third-party developers to sell their custom applications.

5

Why Android?

Listen from the project creators/developers (2.19 min)

- Nick Sears. Co-founder of Android
- Steve Horowitz. Engineering Director
- Dam Morrill. Developer
- Peisun Wu. Engineering Project Manager
- Erick Tseng. Project Manager
- Iliyan Malchev. Engineer
- Mike Cleron. Software Manager
- Per Gustafsson. Graphics Designer.
- etc…

LINK: http://www.youtube.com/watch?v=6rYozIZOgDk&eurl=http://www.android.com/about/&feature=player_embedded

You will hear statements such as:

"…currently it is too difficult to make new products … open software brings more innovation … choices … lower costs … enables the industry to create….more applications such as family planner, my taxes, … understand my wife better, … "

6

What is the Open Handset Alliance?

A consortium of 80+ technology and mobile business companies.

Quoting from **www.OpenHandsetAlliance.com** site (2/25/2012)

" … Today, there are 1.5 billion television sets in use around the world. 1 billion people are on the Internet. But nearly 3 billion people have a mobile phone, making it one of the world's most successful consumer products…

Building a better mobile phone would enrich the lives of countless people across the globe.

The Open Handset Alliance™ is a group of mobile and technology leaders who share this vision for changing the mobile experience for consumers …"

7

Open Handset Alliance Members

Operators	Software Co.	Commercializat.	Semiconductor	Handset Manf
Bouygues Tele China Mobile China Telec. China Unicom KDDI Corp. NTT DoCoMo Softbank Sprint Nextel Telecom Italia Telefónica Telus T-Mobile … Vodafone	Ascender Corp. Borqs eBay Esmertec Google LivingImage NMS Comm. Nuance Comm. PacketVideo SkyPop … SONiVOX	Accenture Aplix Astonishing Tribe Noser Engineering Omron Software Sasken Teleca … Wind River Systems	ARM Atheros Audience Broadcom Corp. CSR Plc. Cypress Freescale Gemalto Intel Corp. Marvell Tech MediaTek MIPS Techn. Nvidia Corp Qualcomm Renesas Corp ST-Ericsson Synaptics Texas Instrum. Via Telecom	ACER ASUS Dell Garmin HTC Kyocera Lenovo Mobile LG Motorola NEC Samsung Samsung Sharp … Sony Ericsson … Toshiba

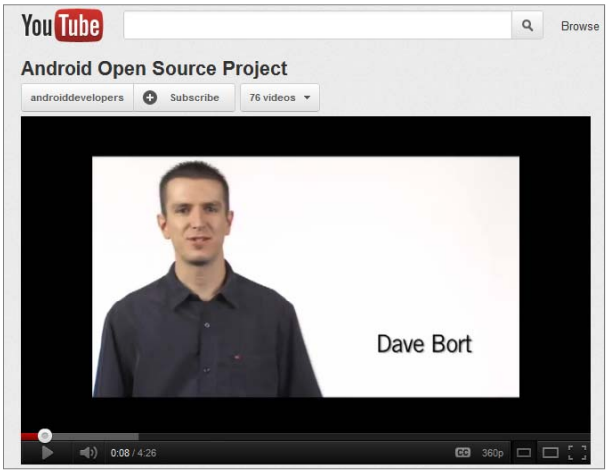
8

See Android Developers

http://www.youtube.com/watch?v=7Y4thikv-OM

Short video (4 min.)

Dave Bort and *Dan Borstein*, members of the Android Open Source Project talk about their experience.



9

The Mobile Revolution

Electronic tools of a typical business warrior

Not so long ago …	Today
1. Phone 2. Pager 3. PDA Organizer 4. Laptop 5. MP3 Portable music player 6. Wired modem 7. No Internet access / limited access	1. Smartphone 2. Laptop (perhaps!)

Tomorrow ?

10

The Mobile Revolution

Dreaming aloud
I want my 2015 Smartphone to be …

1. Phone
2. Pager
3. PDA Organizer
4. High Quality Camera (still & video)
5. Portable music player
6. Portable TV / Video Player / Radio
7. Laptop
8. Play Station
9. GPS / Compass / Navigation (road & inside buildings)
10. Golf Caddy (ball retriever too)
11. Book Reader (I don't read, It reads to me)
12. Electronic key (Car / Home / Office)
13. Remote Control (Garage, TV, …)
14. Credit Card / Driver's License / Passport / Airplane Ticket
15. Cash
16. Cook, house chores
17. Psychologist / Mentor / Adviser
18. Personal trainer
19. Dance instructor
20. ????

11

Android vs. OS Competitors

vs.	1. Apple Inc. 2. Microsoft ~~3. Nokia Symbiam~~ ~~4. Palm & webOS~~ 5. Research In Motion

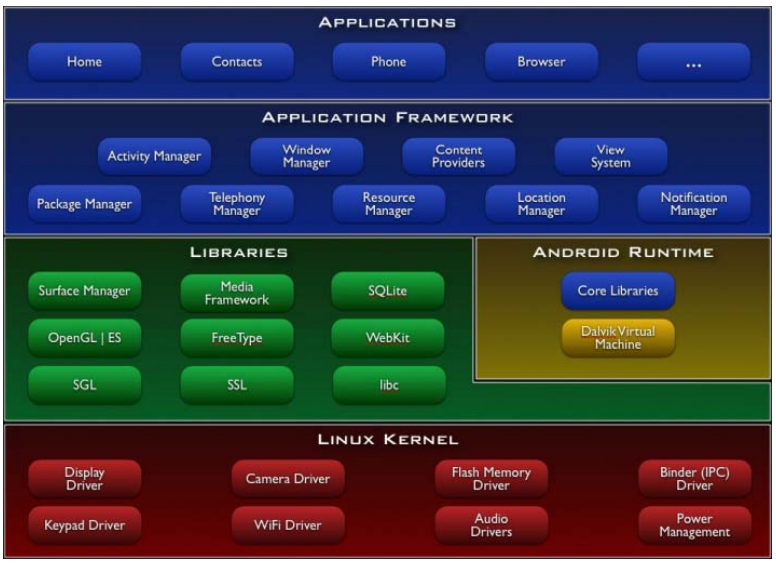
12

Android Software/Hardware Components

- **Dalvik virtual machine**
- **Integrated browser** (WebKit)
- **Graphic Capabilities** (hardware acceleration)
- **SQLite** for structured data storage
- **Media support** (audio/video)
- **GSM Telephony** (hardware dependent)
- **Bluetooth, EDGE, 3G, 4G, and Wi-Fi** (hardware dependent)
- **Camera, GPS, compass, and accelerometer** (hardware dependent)
- **Software Development Tools** & **Application framework**
  (device emulator, debugging, profiling, plugin for the Eclipse IDE, resource managers)

13

Android Components



14

Android Components

Video 1/3:

Android's Architecture

Presented by Mike Cleron, Google Corp. (13 min)
Available at: http://www.youtube.com/watch?v=QBGfUs9mQYY

Android Components

Video 2/3:

Application's Life Cycle

Presented by Mike Cleron, Google Corp. (7 min)
Available at: http://www.youtube.com/watch?v=fL6gSd4ugSI&feature=channel

Android Components

Video 3/3:

Andoid's API

Presented by Mike Cleron, Google Corp. (8 min)
Available at: http://www.youtube.com/watch?v=MPukbH6D-lY&feature=channel

17

Android Application Framework

Video:

Inside the
Android Application Framework
(about 52 min)

Presented by Dan Morrill – Google
At Google Developer Conference
San Francisco - 2008
Available at:
http://sites.google.com/site/io/inside-the-android-application-framework

Android is designed to be fast, powerful, and easy to develop for. This session will discuss the Android application framework in depth, showing you the machinery behind the application framework.

explains the life-cycle of an android apk. very good!

18

Android Support - Education

Video:

Android Development Tools
(about 60 min)

Google 2011 Developer Conference
San Francisco

Presented by

- Xavier Ducrohet, tech-lead for the Android SDK and Developer Tools.
- Tor Norbye, engineer on the Android SDK team working on visual tools for Android development.

LINK: http://www.google.com/events/io/2011/sessions/android-development-tools.html

Android Components

Video:

An Introduction to Android
(about 52 min)

Presented by Jason Chen – Google
At Google Developer Conference
San Francisco - 2008
Available at:
http://www.youtube.com/watch?v=x1ZZ-R3p_w8

Dalvik Virtual Machine

Video (61 min)

Dalvik VM Internals

Presented by Dan Borstein

At Google Developer – 2008

San francisco

Available at:

http://www.youtube.com/watch?v=ptjedOZEXPM

21

Android Intents

- An **Intent** is a request for services.
- An **Intent** is made up of various pieces including:
  - desired *action* or *service*,
  - *data*, and
  - *category* of component that should handle the intent and instructions on how to launch a target activity.

22

Example of Built-In Android Intents

Action	Data
The general action to be performed, such as: ACTION_VIEW, ACTION_EDIT, ACTION_MAIN, etc.	The data to operate on, such as a person record in the contacts database, expressed as a Uri.

Intents

Some examples of Intent's action/data pairs are:

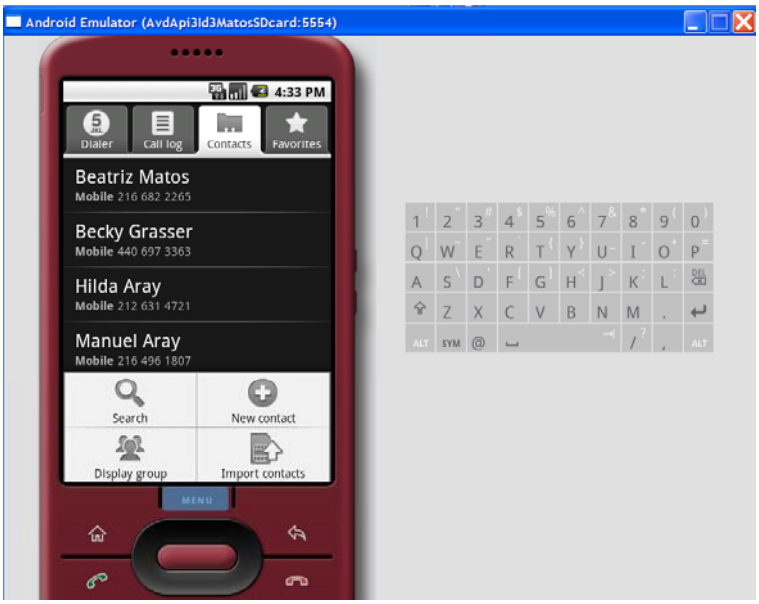
ACTION_VIEW ***content://contacts/1*** -- Display information about the person whose identifier is "1".
ACTION_DIAL ***content://contacts/1*** -- Display the phone dialer with the person filled in.
ACTION_VIEW ***tel:123*** -- Display the phone dialer with the given number filled in
ACTION_DIAL ***tel:123*** -- Display the phone dialer with the given number filled in.
ACTION_EDIT ***content://contacts/1*** -- Edit information about the person whose identifier is "1".
ACTION_VIEW ***content://contacts/*** -- Display a list of people, which the user can browse through.

Example1: Java + Built-in Intent

The following fragment calls an **Intent** whose job is to invoke a built-in task (*ACTION_VIEW*) and explore the *Contacts* available in the phone.

```
Intent myIntent = new Intent(
                  Intent.ACTION_VIEW,
                  Uri.parse("content://contacts/people"));

startActivity(myIntent);
```


Example1: Built-in Intent



Intent uses ACTION_VIEW to see Contacts.

Example1: Built-in Intent

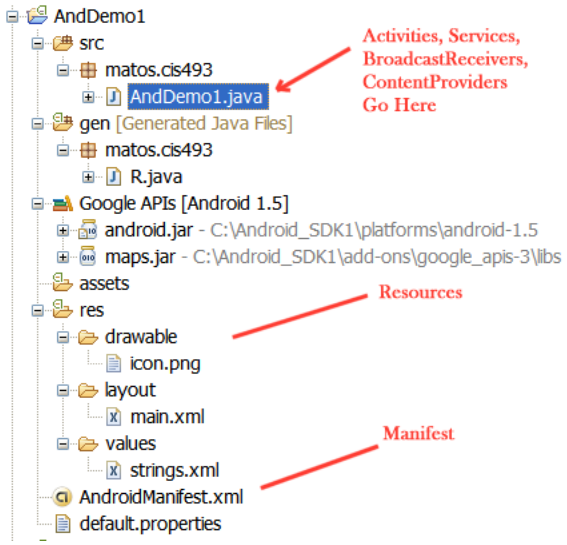
Complete code to see Contacts.

```
public class AndDemo1 extends Activity {
    /** show contact list */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Intent myIntent = new Intent(
                Intent.ACTION_VIEW,Uri.parse( "content://contacts/people"));

        startActivity(myIntent);
    }
}
```

27

Pieces of an Android Application

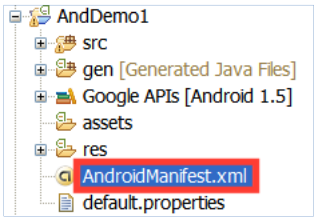
Structure of a typical Android Application



28

Android Manifest xml File

- Every application must have an AndroidManifest.xml file (with precisely that name) in its root directory.
- The manifest presents essential information about the application to the Android system, information the system must have before it can run any of the application's code.



29

Android Manifest xml File

These are the only legal elements; you cannot add your own elements or attributes.

<action>	<permission>
<activity>	<permission-group>
<activity-alias>	<permission-tree>
<application>	<provider>
<category>	<receiver>
<data>	<service>
<grant-uri-permission>	<uses-configuration>
<instrumentation>	<uses-library>
<intent-filter>	<uses-permission>
<manifest>	<uses-sdk>
<meta-data>	

30

Android Manifest xml File

Among other things, the manifest does the following:

- It names the *Java package for the application*. The package name serves as a unique identifier for the application.
- It describes the components of the application — the *activities*, *services*, *broadcast receivers*, and *content providers* that the application is composed of.
- It names the *classes* that implement each of the components and publishes *their capabilities* (for example, which Intent messages they can handle). These declarations let the Android system know what the components are and under what conditions they can be launched.
- It determines which processes will *host application components*.
- It declares which *permissions* the application must have in order to access protected parts of the API and interact with other applications.
- It also declares the permissions that others are required to have in order to interact with the application's components.
- It lists the *Instrumentation* classes that provide profiling and other information as the application is running. These declarations are present in the manifest only while the application is being developed and tested; they're removed before the application is published.
- It declares the minimum level of the *Android API* that the application requires.
- It lists the *libraries* that the application must be linked against.

31

Android Manifest xml File

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
      package="matos.earthquake"
      android:versionCode="1"
      android:versionName="1.0.0">
    <application android:icon="@drawable/yellow_circle" android:label="@string/app_name">


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

        <activity android:name=".SatelliteMapping">  </activity>

        <service  android:name="AndQuakeService" android:enabled="true" >
        </service>

        <receiver android:name="AndQuakeAlarmReceiver" >
           <intent-filter>
             <action
                  android:name = "ALARM_TO_REFRESH_QUAKE_LIST"/>
           </intent-filter>
         </receiver>
    </application>

    <uses-library android:name="com.google.android.maps" />
    <uses-permission android:name="android.permission.INTERNET" />

</manifest>
```

32

Example2. Currency converter

Implementing a simple currency converter:

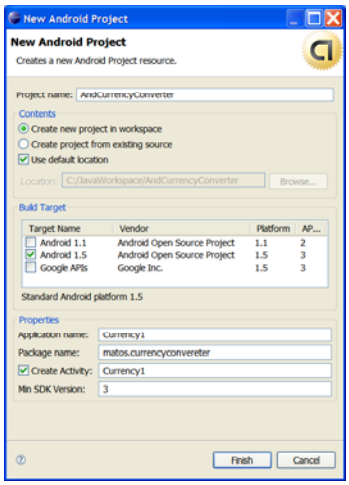
USD – Euro – Colon (CR)

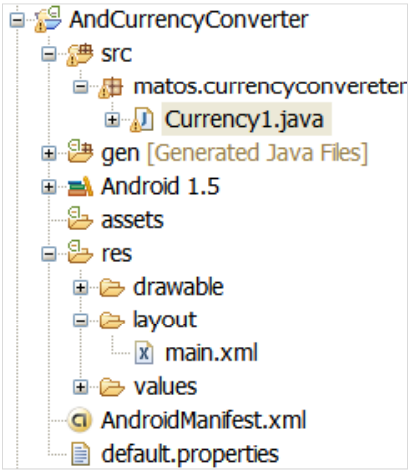
Note. Naive implementation using the rates

1 Costa Rican Colon = 0.001736 U.S. dollars

1 Euro = 1.39900 U.S. dollars

Example2. Currency converter





Example2. Currency converter

```
package matos.currencyconvereter;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class Currency1 extends Activity {
      // naive currency converter from USD to Euros & Colones
     final double EURO2USD = 1.399;
     final double COLON2USD = 0.001736;

     // GUI widgets
     Button btnConvert;
     Button btnClear;
     EditText txtUSDollars;
     EditText txtEuros;
     EditText txtColones;
```

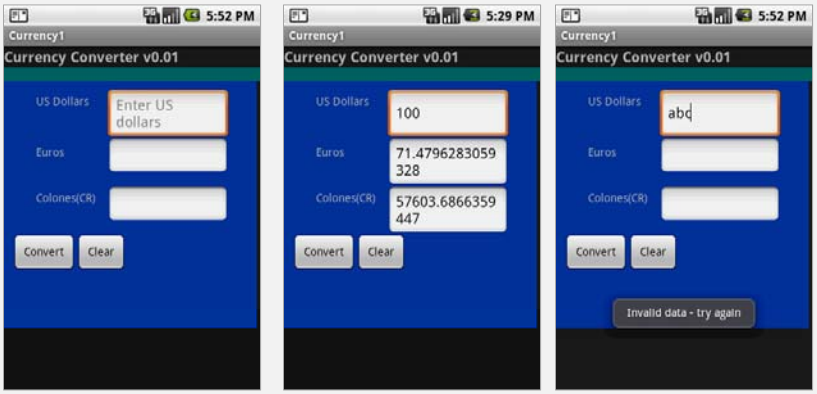
Example2. Currency converter

```
@Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        // bind local controls to GUI widgets
        txtUSDollars = (EditText)findViewById(R.id.txtUSDollars);
        txtUSDollars.setHint("Enter US dollars");
        txtEuros = (EditText)findViewById(R.id.txtEuros);
        txtColones = (EditText)findViewById(R.id.txtColones);

      // attach click behavior to buttons
        btnClear = (Button)findViewById(R.id.btnClear);
        btnClear.setOnClickListener(new OnClickListener() {
             // clear the text boxes
           @Override
           public void onClick(View v) {
                 txtColones.setText("");
                 txtEuros.setText("");
                 txtUSDollars.setText("");
           }
        });
```

Example2. Currency converter

```
        // do the conversion from USD to Euros and Colones
         btnConvert = (Button) findViewById(R.id.btnConvert);
         btnConvert.setOnClickListener(new OnClickListener() {
           @Override
           public void onClick(View v) {
               try {
                   String usdStr = txtUSDollars.getText().toString();
                   double usd = Double.parseDouble( usdStr );
                   String euros = String.valueOf( usd / EURO2USD );
                   String colones = String.valueOf( usd / COLON2USD );
                   txtEuros.setText(euros);
                   txtColones.setText(colones);
               } catch (Exception e) {
                   Toast.makeText(v.getContext(), "Invalid data - try again"
            ,                   Toast.LENGTH_SHORT).show();
               }
          }
          });// setOnClick...
     }// onCreate

}// class
```

Example2. Currency converter



Example2. Currency converter

Resource: res/ layout/main.xml (1/2)

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
android:id="@+id/widget47"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:orientation="vertical"
xmlns:android="http://schemas.android.com/apk/res/android"
>

<TextView
android:id="@+id/caption1"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:text="Currency Converter v0.01"
android:textSize="18sp"
android:textStyle="bold"
>
</TextView>

<TextView
android:id="@+id/greenFiller1"
android:layout_width="fill_parent"
android:layout_height="wrap_content"
android:background="#ff006666"
>
</TextView>

<AbsoluteLayout
android:id="@+id/absLayout"
android:layout_width="316px"
android:layout_height="308px"
android:background="#ff003399"
>

<TextView
android:id="@+id/usdCaption"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="US Dollars"
android:layout_x="40px"
android:layout_y="15px"
>
</TextView>

<EditText
android:id="@+id/txtUSDollars"
android:layout_width="150px"
android:layout_height="wrap_content"

android:layout_x="130px"
android:layout_y="10px"
>
</TextView>
```

Example. Currency converter

Resource: res/ layout/main.xml (2/2)

```
<EditText
android:id="@+id/txtEuros"
android:layout_width="150px"
android:layout_height="wrap_content“
android:layout_x="130px"
android:layout_y="70px"
>
</EditText>

<TextView
android:id="@+id/colonCaption"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Colones(CR)"
android:layout_x="40px"
android:layout_y="135px"
>
</TextView>
<EditText
android:id="@+id/txtColones"
android:layout_width="150px"
android:layout_height="wrap_content“
android:layout_x="130px"
android:layout_y="130px"
>
</EditText>

<Button
android:id="@+id/btnConvert"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text=" Convert "
android:layout_x="10px"
android:layout_y="190px"
>
</Button>

<Button
android:id="@+id/btnClear"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text=" Clear "
android:layout_x="90px"
android:layout_y="190px"
>
</Button>

</AbsoluteLayout>

</LinearLayout>
```

Example. Currency converter

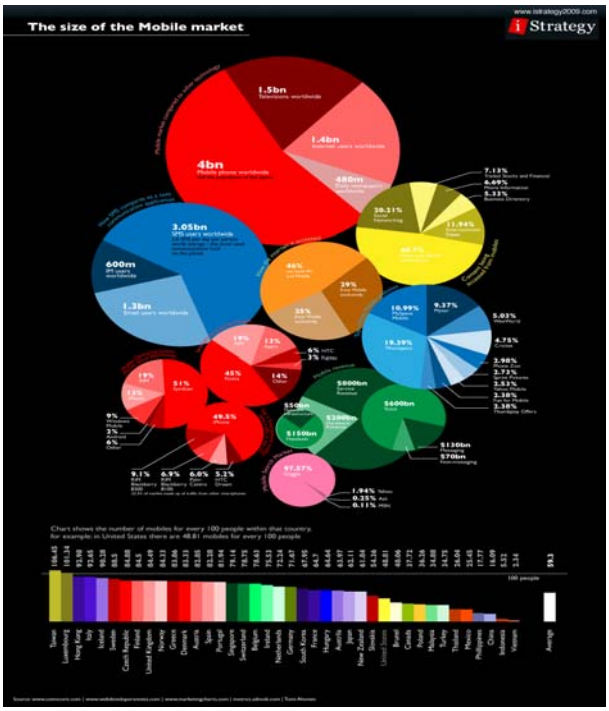
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
      package="matos.currencyconvereter"
      android:versionCode="1"
      android:versionName="1.0">
    <application android:icon="@drawable/icon"
                 android:label="@string/app_name">
        <activity android:name=".Currency1"
                  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>
```

APPENDIX

The Size of the Mobile Market

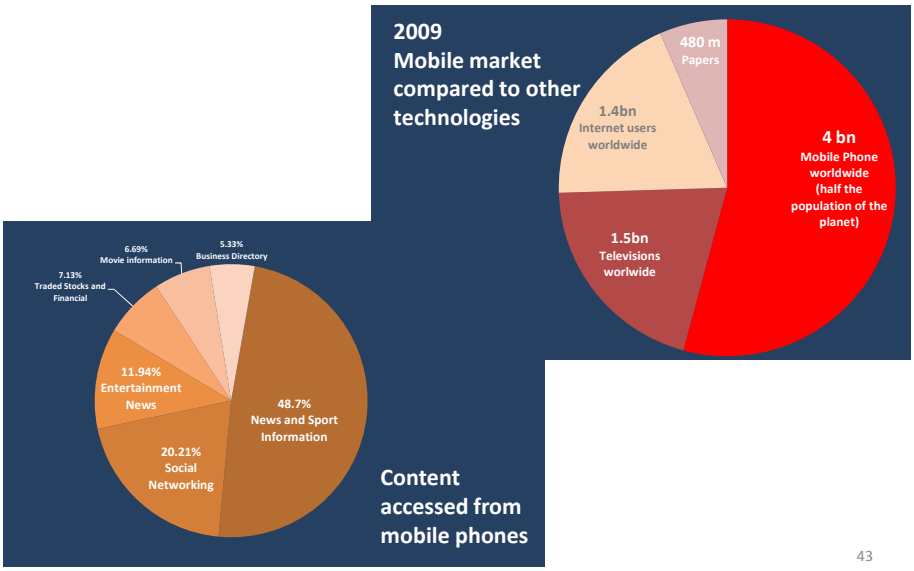
Reference:

http://gizmodo.com/5489036/cellphone-overshare



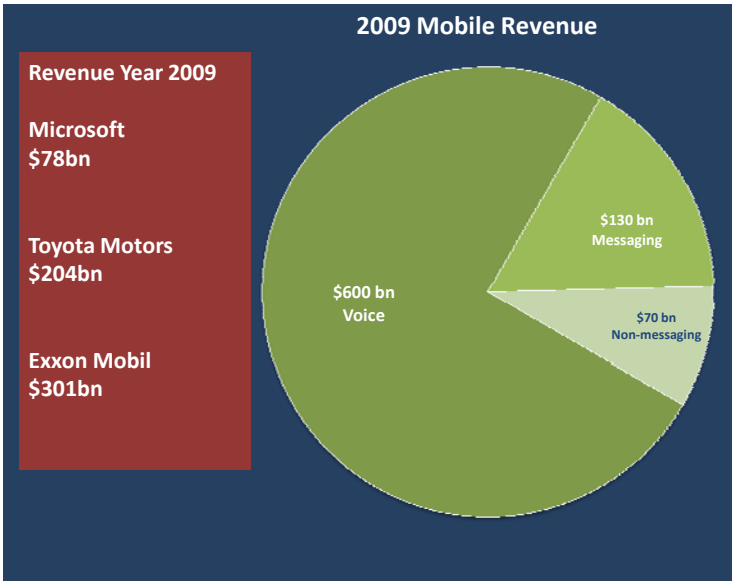
The Size of the Mobile Market – 2009

Extracted from: http://gizmodo.com/5489036/cellphone-overshare



The $ize of the Mobile Market – 2009

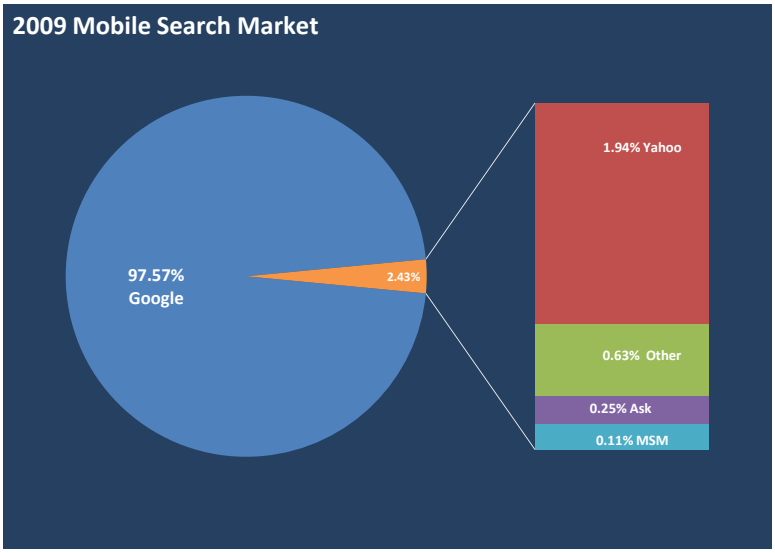
Extracted from:
http://gizmodo.com/5489036/cellphone-overshare
http://www.microsoft.com/investor/reports/ar09/10k_fr_bal.html
Exxon Mobil 2009 Summary Annual Report
2010 Toyota Annual Report (pp 12)



Revenue is the collective amount of income made by a company (usually from the sales of goods & services)

The Size of the Mobile Market – 2009

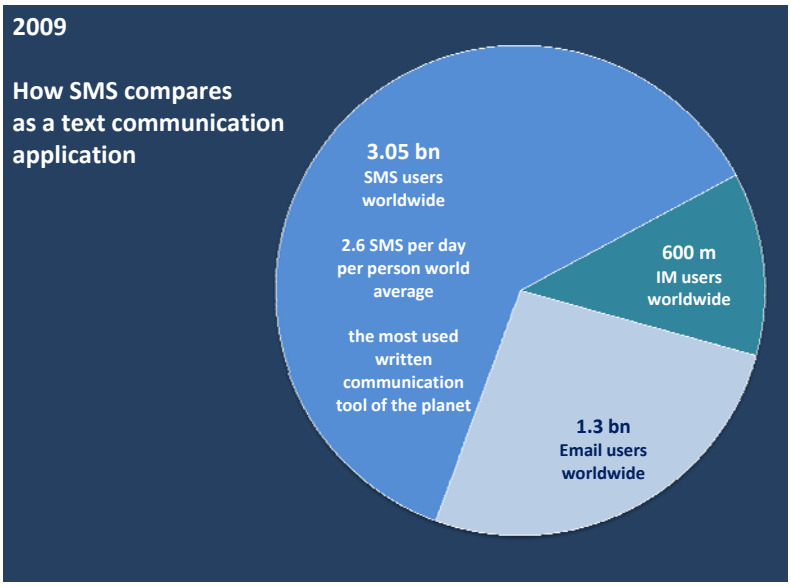
Extracted from: http://gizmodo.com/5489036/cellphone-overshare



45

Appendix. The Size of the Mobile Market – 2009

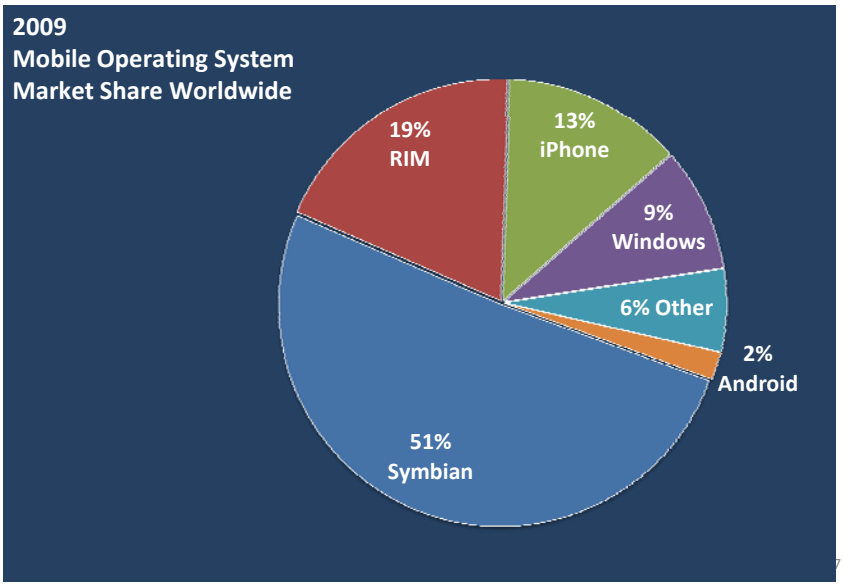
Extracted from: http://gizmodo.com/5489036/cellphone-overshare



46

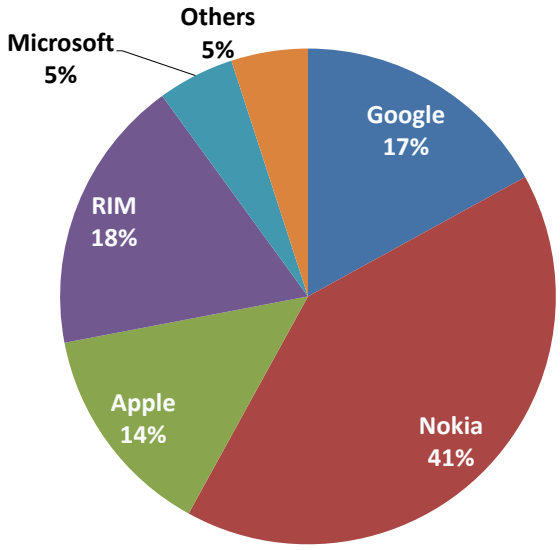
The Size of the Mobile Market – 2009

Extracted from: http://gizmodo.com/5489036/cellphone-overshare



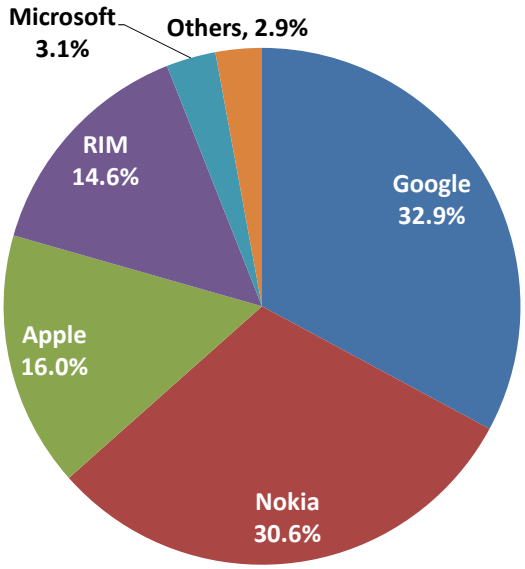
The Size of the Mobile Market – Q2 / 2010

Extracted from: http://www.businessinsider.com/android-iphone-market-share-2010-8



The Size of the Mobile Market – Q4 / 2010

Extracted from http://www.canalys.com/pr/2011/r2011013.pdf

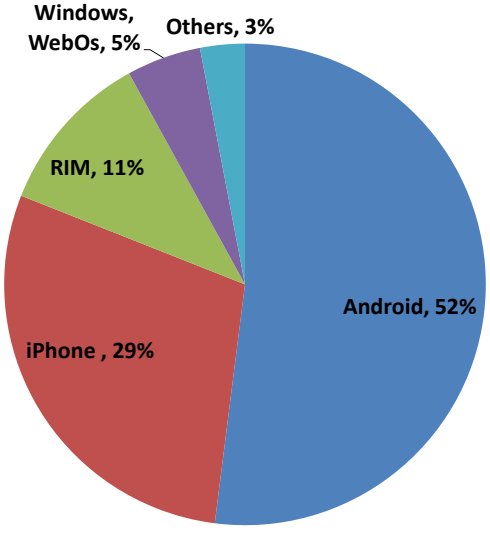


Combined sales in Q4
300 million units

49

The Size of the Mobile Market – Q2 / 2011

Extracted from http://www.channelinsider.com/c/a/Messaging-and-Collaboration/Android-Smartphone-Marketshare-Grows-NPD-Report-469609/



50

New Products for the 2011 Year

Motorola Atrix 4G (ATT Store)
Dual-core processor - 1 GB RAM

Tablets

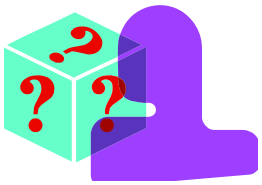
51

Cell-Phone Diffusion

Dr. Lyza Lyth
Mama Justine & Children

Mount Kilimangaro
Tanzania, October 2010

52



Thanks for being here

Questions?

Bibliography:

1. Android Developers. http://developer.google.com/index.html
2. Professional Android Application Development by Reto Meier ISBN: 978-0-470-34471-2. Wrox Publications, 2008.
3. Unlocking Android by Frank Ableson, Charlie Collins, and Robi Sen. ISBN 978-1-933988-67-2. Manning Publications, 2009.
4. Professional Android 2 Application Development (Wrox Programmer to Programmer) by Reto Meier. ISBN-10: 0470565527. Wrox Pub. 2010.
5. The Busy Coder's Guide to Advanced Android Development by Mark Murphy. ISBN ISBN: 978-0-9816780-5-4. CommonsWare Pub. 2012.
6. Android Programming Tutorials by Mark Murphy. ISBN ISBN: 9 ISBN: 978-0-9816780-7-8. CommonsWare Pub. 2011.