Android Development
Introduction

In this chapter, you will be introduced to mobile development and its future.
Desktop VS Mobile

- Internet usage

![Graph showing the comparison of desktop and mobile internet usage from 2007 to 2015, with a peak in 2014.](image-url)
1,500,000 Android phones being activated every day
Android statistics

- 600 Manufactures, 4000 million Android devices
Android statistics

- 100 billion app installs from the Android Market
Easy way into the market
Smartphones (2014)

- Android: 78%
- iOS: 18%
- Windows Phone: 3%
- BlackBerry: 3%
- Other: 1%
HTML vs Native mobile applications

**Pros**

- No need for installation
- Multi platform *

**Cons**

- Security issues
- Not the same native experience
- Not in app store

We can use hybrid application (Ynet)
Brief History about Android

- In 2005 Google Acquired a startup called Android
- In 2008 Google released version 1.0
- Because it is free it is used by many companies
- Android is used not only for phones
Android versions

- ... 
- Android 4.0 Ice Cream Sandwich (Combines 2.3 and 3.0)
- Android 4.1 & 4.2 & 4.3-> Jelly bean
- Android 4.4 -> KitKat
- Android 5.0 -> Lollipop
- Android 6.0 -> ?
Setup Environment

In this chapter, you will learn how to setup the Android development environment.
Android development site: http://developer.android.com

Development site tour
How to install Android Environment

- Android Studio IDE
- Android SDK tools
- Android 5.0 (Lollipop) Platform
- Android 5.0 emulator system image with Google APIs

Download Android Studio for Windows

- System Requirements
- Other Download Options
- Migrating to Android Studio
- Take a Survey
Android SDK Manager

Android SDK tour
About Device Manager

AVD tour
Preparing the phone (optional and recommended)

- Install the phone USB driver
- Enable USB debugging
Where to ask questions?

- Android developers spend most of their time at: www.stackoverflow.com
- The more stupid the question is, the faster you will get an answer
You can use a better Android Emulator: https://www.genymotion.com/
In this chapter, you will learn about the basic architecture of the Android OS
Linux Kernel

- Based on Linux for core system services (Security, Memory management, network stack, drivers)
- Connects with the Hardware
Applications process

- Each application is a different User
- Each application has it’s own process
- C/C++ libraries
  - System C library
  - Media library
  - 3D libraries
  - SQLite
Android Runtime

- Every App runs on its own process
- The Dalvik VM
Application Framework

- Basic Android code
  - Activity Manager
  - Window Manager
  - Notification Manager
  - Location Manager

Sending SMS example
SmsManager sms = SmsManager.getDefault();
sms.sendTextMessage(phoneNumber, null, message, sentPI, deliveredPI);
Applications

- Written in Java
- The output file is .apk file
- Each application runs in its own process
- Applications contains:
  - Manifest file
  - Resource files
  - Components (Activities, Services, Content Providers, Broadcast receivers)

Tour - Create a new project
Hello Sheep

Ready for your first Android application? In this chapter you will create your android “hello world” application
private static final String TAG = "SheepActivity";
Log.d(TAG, "index=" + i);

<table>
<thead>
<tr>
<th>Constants</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>int ASSERT</td>
<td>Priority constant for the println method.</td>
</tr>
<tr>
<td>int DEBUG</td>
<td>Priority constant for the println method; use Log.d.</td>
</tr>
<tr>
<td>int ERROR</td>
<td>Priority constant for the println method; use Log.e.</td>
</tr>
<tr>
<td>int INFO</td>
<td>Priority constant for the println method; use Log.i.</td>
</tr>
<tr>
<td>int VERBOSE</td>
<td>Priority constant for the println method; use Log.v.</td>
</tr>
<tr>
<td>int WARN</td>
<td>Priority constant for the println method; use Log.w.</td>
</tr>
</tbody>
</table>
First Exercise

Now you do it!

Ex1:
- Install the android environment
- Create a new project and log out your hello message in the onCreate()

Advanced:
- Find how to log a message when the screen turns off (or user switch to other application)
Activities

- Activity is a single screen with UI
- Each Application can have several activities

How to create activity:
- Create a subclass of Activity
- Declare the activity in the manifest
- Start the Activity (Intent)

The easy way:
File->New->Activity (from the Android studio menu)

Tour – Create a new Activity