



Programmazione Android

Giovanni Perbellini, Stefano Cordibella
Università di Verona – EDALab S.r.l.

Agenda

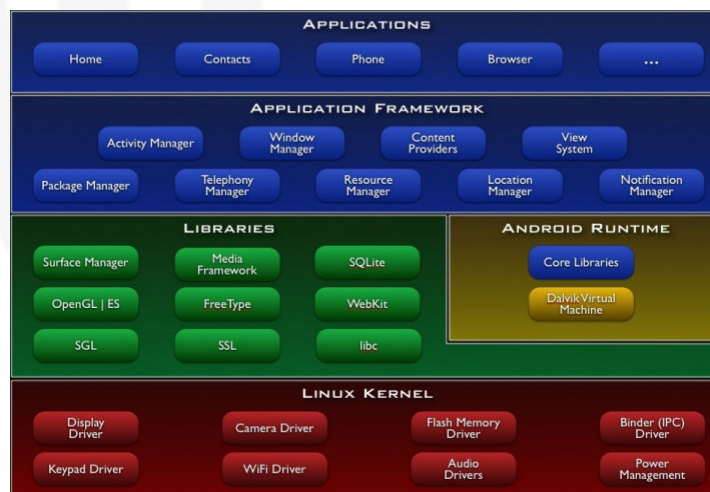
- Introduzione Android
 - Overview
 - Ambiente di sviluppo
- Esempi
 - Helloworld
 - Weather

Cos'è Android?

- Stack software che include:
 - Sistema operativo
 - Middleware
 - Applicazioni
- Android SDK
 - Fornisce gli strumenti e le API per sviluppare applicazioni su terminali mobili Android utilizzando il linguaggio di programmazione Java

3

Architettura Android



4

Application

- Email
- Calendar
- Agenda
- Maps
- Browser
- SMS program
- Etc.

5

Application framework (I)

- Insieme di componenti, API e servizi utili per lo sviluppo di nuove applicazioni.
 - Stesse API usate per lo sviluppo delle core applications.
- Si sfrutta il concetto del riutilizzo dei componenti
 - ogni applicazione può usare le potenzialità/caratteristiche di altre applicazioni

6

Application framework (II)

- Include i seguenti servizi:
 - View
 - Lists, grids, buttons, text boxes, web browser, etc.
 - Content providers
 - Permette alle applicazioni di accedere ai dati di altre applicazioni (e.g., Agenda) oppure di condividere i propri dati ad altre applicazioni.
 - Resource manager
 - Accesso a risorse del tipo: file esterni (XML, PNG, JPEG) inclusi nell'applicazione a build-time.
 - Notification manager
 - Per fornire alle applicazioni uno strumento di notifica
 - Activity manager
 - Gestisce il ciclo di vita di ogni applicazione

7

Libraries

- Insieme di librerie C/C++ usate dai vari componenti di Android.
- Esportate allo sviluppatore attraverso Android application framework.
- Alcuni esempi:
 - **System C library**: implementazione della libc per embedded system.
 - **SQLite**: DB embedded.
 - **Media libraries**: gestione di componenti audio video.
 - Etc.

8

Android runtime

- Include le librerie che forniscono le funzionalità principali di Java.
- Dalvik Virtual Machine
 - Esegue file nel formato .dex (Dalvik Executable)
- Possibilità di caricare più VM in modo efficiente.

9

Linux kernel

- Sfrutta Linux kernel 2.6 per i servizi principali come:
 - Security, memory management, process management, network stack, driver model.

10

Ambiente di sviluppo (I)

- Android SDK
 - Contiene API, compilatore, debugger, emulatore (Android Virtual Device), etc.
 - Aggiungere Android platform (1.6 o 2.0) attraverso AVD Manager
- Eclipse
 - IDE per lo sviluppo di applicazioni
- Eclipse Plugin
 - ADT (Android Development Tools)
 - Permette di creare un progetto Android
 - Aggiunge componenti di Android Framework API
 - Debug applicazioni tramite Android SDK tools
 - Crea l'APK (Android Package) file per distribuire l'applicazione

11

Ambiente di sviluppo (II)

- Possibilità di creare un progetto Android senza Eclipse
- Android SDK include il tool “*android*” per creare tutto il codice sorgente, directory, file *ant* per la compilazione.
- Esempio di creazione del progetto HelloAndroid

```
android create project \
  --package com.android.helloandroid \
  --activity HelloAndroid \
  --target 2 \
  --path <path-to-your-project>/HelloAndroid
```

12

Esempio: Helloworld (I)

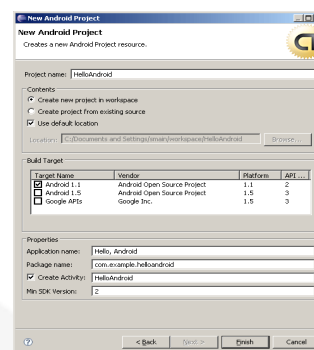
- Creare il nuovo progetto:
 - File -> New -> Project...
 - Selezionare Android -> Android Project -> Next



13

Esempio: Helloworld (II)

- Riempire i campi:
 - Project Name -> HelloAndroid
 - Selezionare la piattaforma desiderata
 - Application Name : HelloAndroid
 - Package Name: org.edalab
 - Create Activity: Hello
- Finish



14

Esempio: Helloworld (III)

- Editare il file in src -> org.edalab -> Hello.java aggiungendo:

```
// We want to view some very simple text, so we need a TextView
TextView tv = new TextView(this);
// Put some text to the newly created TextView
tv.setText("Hello Android!!!!\n");
// Tell our App to display the textView
this.setContentview(tv);
```

- eliminare la riga:

```
setContentview(R.layout.main);
```

15

Esempio: Helloworld (IV)

- A questo punto si possono aggiungere elementi grafici attraverso:
 - res -> layout -> main.xml .
- Esistono tool più evoluti presenti nella stessa sdk come ad esempio layoutopt.

16

Esempio: Weather (I)

- Applicazione per la visualizzazione di informazioni metereologiche dato come input la città.
- Contenuti:
 - Reperimento di informazioni da internet
 - Interazione con le Google Weather API
 - Parsing di file XML (utilizzando SAXParser)
 - Personalizzare la parte grafica estendendo LinearLayout

17

Esempio: Weather (II)

- Importare il progetto:
 - File -> Import -> General -> Existing project into workspace
 - Selezionare Next
 - Selezionare Select archive file e quindi Browse
 - Puntare all'archivio WeatherForecast.tar.gz
 - Finish

18

Esempio: Weather (III)

- Impostare la piattaforma desiderata:
 - Project -> Properties -> Android
 - Selezionare la piattaforma 2.1
 - Apply e quindi OK
 - Selezionare il progetto con il tasto destro -> Android Tools -> Fix Project Properties
- Ora è possibile eseguire l'applicazione.

19

Esempio: Weather (IV)

- Effettuare una HTTP Get (WeatherForecast.java):

```

/* Use HttpClient to deal with the URL */
HttpClient httpClient = new DefaultHttpClient();
HttpGet httpget = new HttpGet(queryString.replace(" ", "%20"));
// create a response handler
ResponseHandler<String> responseHandler = new
    BasicResponseHandler();
String responseBody = httpClient.execute(httpget, responseHandler);

```

20

Esempio: Weather (V)

- Interazione con le Google Weather API:
 - Formattazione della richiesta http secondo il formato: <http://www.google.com/ig/api?weather=QUERY> dove QUERY ha il formato Città,Nazione; ad es: <http://www.google.com/ig/api?weather=Verona,Italy>

21

Esempio: Weather (VI)

- Parsing di file XML (SAXParser):

```

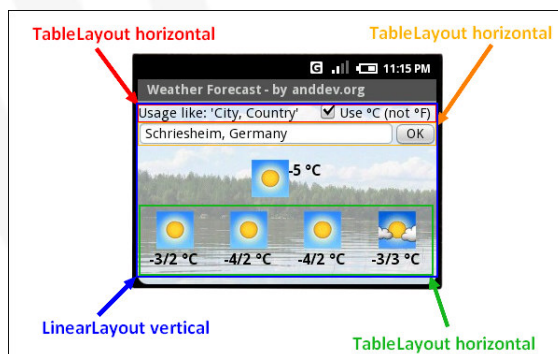
/* Get a SAXParser from the SAXParserFactory. */
SAXParserFactory spf = SAXParserFactory.newInstance();
SAXParser sp = spf.newSAXParser();
/* Get the XMLReader of the SAXParser we created. */
XMLReader xr = sp.getXMLReader();
/* Create a new ContentHandler and apply it to the XML-Reader */
/* GoogleWeatherHandler is a SAXHandler capable of extracting
   information out of the xml-data returned by the Google Weather
   API. */
GoogleWeatherHandler gwh = new GoogleWeatherHandler();
xr.setContentHandler(gwh);

```

22

Esempio: Weather (VII)

- Personalizzazione della parte grafica:
 - Creazione della classe `SingleWeatherInfoView`



23

Esempio: Weather (VIII)

- Uso dell'applicativo Hierarchy Viewer:
 - Applicazione Java che permette di navigare la struttura degli elementi grafici a run time;
 - Si avvia lanciando `hierarchyviewer` da terminale con l'emulatore attivo;
 - Due possibilità di visualizzazione:
 - Layout View
 - Pixel Perfect View

24