

Introduzione

No questions... no answers...

Men in Black III, 2012

Negli ultimi anni, Android si è affermata come una delle principali piattaforme per lo sviluppo di applicazioni mobili, dividendosi quasi tutto il mercato con l'ormai acerrimo rivale iOS di Apple. Sebbene le due piattaforme si rincorrono dal punto di vista delle funzionalità, si differenziano sicuramente dal punto di vista di quella che si chiama *apertura*, o meglio con il concetto di *open*. Mentre la piattaforma di Apple è chiusa (non personalizzabile), il produttore è unico e le tipologie di dispositivi sono in numero limitato, Android rappresenta la collezione di un numero elevatissimo di versioni diverse su dispositivi diversi realizzati da *vendor* diversi. Potremmo discutere per centinaia di pagine su vantaggi e svantaggi di queste due filosofie; sicuramente quello che stiamo vedendo è un proliferare di specializzazioni di Android, culminata proprio in questi giorni (siamo a inizio aprile 2013) dall'ufficializzazione della piattaforma Facebook Home. Si tratta infatti, a grandi linee, di un'applicazione personalizzata di *Home* la quale permette l'avvio delle normali applicazioni Android oltre che l'accesso a funzionalità caratteristiche del social network Facebook.

Indipendentemente da cosa e da come la si pensi, Android rappresenta sicuramente una piattaforma che lo sviluppatore, professionista o meno, deve imparare a conoscere per non rimanere indietro nello sviluppo delle applicazioni mobili, e questo è l'obiettivo del presente testo.

A differenza delle versioni precedenti, ho deciso di seguire un approccio completamente diverso. Sebbene io rimanga sempre dell'opinione che "i fondamenti sono fondamentali" e che gli aspetti teorici siano necessari alla comprensione di un linguaggio o piattaforma, ho deciso di cambiare e seguire un metodo molto pratico che porterà il lettore, passo dopo passo, alla realizzazione di un'applicazione che abbiamo chiamato UGHO (*User Generated HOroscope*) e che sarà un ottimo pretesto per affrontare la maggior parte degli aspetti relativi allo sviluppo di un'applicazione Android.

Il Capitolo 1 ci permetterà di scaldare i motori e prepararci a partire affrontando tutti i concetti che sono necessari allo sviluppo con questa piattaforma. Vedremo quindi che cos'è Android, che relazione ha con Java e descriveremo i principi fondamentali di questo linguaggio. Non sarà un manuale di Java ma una descrizione, molto pragmatica, di quello che ci sarà utile successivamente nella creazione delle applicazioni. In particolare vedremo che cos'è un *package* e i principali concetti di *object orientation*. Ci occuperemo quindi del *delegation model*, ovvero del meccanismo alla base della gestione degli eventi, di *inner class* e quindi di *generics*. L'obiettivo di questo nostro primo capitolo, che gli esperti

Java potranno eventualmente saltare nella prima parte, è quello di preparare il lettore al viaggio verso la realizzazione di UGHO che, oltre che essere un'applicazione Android, sarà anche qualcosa di divertente.

Il Capitolo 2 è dedicato all'installazione di Android Studio, ovvero dell'IDE che andremo a utilizzare per lo sviluppo. La procedura di installazione è qualcosa su cui Google interviene spesso; nelle precedenti versioni del testo ho dovuto modificare questa parte un numero incredibile di volte. Per questo motivo, la vedremo molto velocemente, rimandando alla documentazione ufficiale per i dettagli. Al termine di questo capitolo avremo capito come creare un progetto e come utilizzare il tool di build che si chiama Gradle. Nel Capitolo 3 iniziamo a fare sul serio, affrontando quelli che sono i concetti di base di ogni applicazione Android. Con il pretesto di creare il flusso di navigazione di UGHO vedremo infatti che cosa sono le *activity* e come questi componenti collaborano tra loro per la visualizzazione delle schermate e il passaggio da una all'altra. Vedremo che cosa sono le risorse e come queste possono essere utilizzate per gestire un numero sempre maggiore di dispositivi diversi attraverso il concetto di *qualificatore*. Affronteremo quindi il concetto di *task* e creeremo i primi layout, di cui impareremo a gestire le preview attraverso l'apposito tool di Android Studio. Per quello che riguarda le *activity*, vedremo nel dettaglio il loro ciclo di vita per poi dedicarci allo studio di un altro concetto fondamentale in Android, ovvero gli *intent* insieme agli *intent filter*. Si tratterà del capitolo più importante e impegnativo del libro, che ci metterà però nelle condizioni di scrivere già alcune applicazioni anche se semplici.

In precedenza abbiamo accennato all'annoso problema di Android che per alcuni è invece una sua caratteristica vincente, ovvero la possibilità di essere eseguito su dispositivi molto diversi tra loro come caratteristiche e soprattutto come dimensioni del display. Il caso tipico è dato dalla possibilità di creare applicazioni che in altri ambiti si dicono *universali*, cioè eseguibili sia su smartphone sia su tablet sfruttando le caratteristiche di ognuno attraverso l'utilizzo di opportuni layout. Il Capitolo 4 è proprio dedicato a questo, e tratta il concetto fondamentale di *fragment* come componente riutilizzabile di un'interfaccia grafica.

I Capitoli 5 e 6 sono un approfondimento di che cos'è una *view*, un `ViewGroup` e in particolare come si utilizzano le `ListView` come elemento fondamentale di ogni layout. In particolare vedremo nel dettaglio che cos'è un *adapter* e quali sono le implementazioni messe a disposizione dall'SDK. Vedremo anche come realizzare un componente personalizzato relativamente complesso utilizzando sia un approccio che abbiamo definito di alto livello (come composizione di altri esistenti) sia uno di basso livello (uso di strumenti di disegno su *Canvas*). Nello specifico realizzeremo un calendario da utilizzare all'interno di un layout o come finestra di dialogo.

Il Capitolo 7 è invece dedicato a un componente molto importante in quanto presente in tutti i pattern UI suggeriti da Google, ovvero l'`ActionBar`. Si tratta di un componente disponibile solamente dalla versione 3.0 della piattaforma che richiede una libreria di terze parti di nome *ActionBarSherlock* (ABS) per poter essere utilizzato con le versioni precedenti. In questo capitolo vedremo come utilizzare l'`ActionBar` standard o come integrare il framework ABS in Android Studio e come utilizzarlo.

Il Capitolo 8 è di fondamentale importanza in quanto tratta tutti gli strumenti che Android ci mette a disposizione per la gestione della persistenza. Descriveremo inizialmente come si gestiscono le `SharedPreferences` fino alla realizzazione di un'interfaccia personalizzabile dei *Settings* dell'applicazione. Dopo una veloce descrizione di come vengano gestiti i file,

vedremo come sia possibile interagire con un database relazionale e più precisamente SQLite. Descriveremo non solo le principali classi ma soprattutto i principali pattern di utilizzo, compresi quelli che riguardano una possibile personalizzazione del `Cursor`. Concluderemo questo fondamentale capitolo con la descrizione e realizzazione di un *content provider*, che vedremo essere molto importante anche in relazione alle funzioni di gestione degli account, oltre ad avere un ruolo cruciale anche nel meccanismo di *intent resolution*. Come avremo occasione di sottolineare più volte, la realizzazione di un'applicazione che risponde in modo reattivo alle azioni dell'utente è una priorità assoluta. È quindi fondamentale padroneggiare tutti i meccanismi di gestione asincrona delle operazioni e in particolare saper gestire il multithreading. Il Capitolo 9 è dedicato a tutti i meccanismi alla base dei processi e dei servizi. Dopo un'introduzione sul concetto di *thread* e *task*, descriveremo nel dettaglio cosa sono gli *handler* e i *looper* in relazione ai possibili scenari di interazione con il thread principale di gestione della UI. Da qui vedremo cos'è un `AsyncTask` e come utilizzarlo in contesti diversi. La seconda parte del capitolo è quindi dedicata a un insieme di strumenti che hanno subito vari miglioramenti nelle ultime versioni, ovvero le *notifiche*; ci dedicheremo poi alla descrizione dei componenti che, insieme alle *activity*, hanno assunto sempre più importanza in tutte le architetture Android: i *service*. A tale proposito realizzeremo due importanti esempi di utilizzo di un servizio nelle due modalità: *Started* e *Bound*. Descriveremo infine l'utilizzo di un *intent service* che sarà utile in diversi casi, come vedremo negli ultimi capitoli.

Il Capitolo 10 è dedicato a due argomenti di fondamentale importanza, ovvero la sicurezza e il networking. Dopo una descrizione di quello che è il *Security Model* di Android, vedremo come eseguire delle richieste HTTP sia attraverso il classico meccanismo che utilizza l'`HttpClient` di Apache sia attraverso il nuovo framework di nome Volley, che impareremo a integrare all'interno di Android Studio.

I Capitoli 11 e 12 sono dedicati a due argomenti in un certo senso slegati dal resto ma che permettono di arricchire le applicazioni di due funzionalità di effetto, ovvero le animazioni e i widget. Per quello che riguarda le animazioni descriveremo sia la modalità classica sia quella che utilizza il nuovo framework chiamato *Property Animator*. Per quello che riguarda i widget, ci occuperemo sia della semplice visualizzazione di un'informazione nella *Home* del dispositivo sia della visualizzazione di un insieme più complesso di informazioni all'interno di un *content provider*. Concluderemo il capitolo descrivendo come sia possibile, nelle ultime versioni di Android, creare dei widget da inserire nella schermata di *lock* del dispositivo, ovvero quella che solitamente ospita l'orologio.

Il Capitolo 13 tratta un argomento che di questi tempi sta assumendo un'importanza sempre maggiore, ovvero la gestione degli account e della sincronizzazione. Nella prima parte vedremo come funziona il sistema basato sull'utilizzo della classe `AccountManager` per la gestione degli account. Vedremo anche come creare un account per la nostra applicazione UGHO utilizzando l'SDK di Facebook, sfruttandone i sistemi di autenticazione e soprattutto le informazioni in esso contenute. Concluderemo quindi il capitolo affrontando il problema della sincronizzazione delle informazioni di un *content provider* con altrettante remote attraverso la definizione di un `SyncAdapter`.

L'ultimo capitolo del testo, il 14, tratta un argomento molto interessante che ci ha permesso di introdurre i *Google Play Services* annunciati all'ultima Google IO del 2013, ovvero le *Google Maps v2.0*. In questo capitolo impareremo a importare queste API e a utilizzarle per la realizzazione di un'applicazione che ci permetta di gestire quelli che si chiamano *Geofence*, che altro non sono che zone per le quali il `LocationManager` genera degli eventi

in occasione dell'ingresso o uscita dalle stesse. Nello stesso capitolo accenneremo all'utilizzo della funzionalità chiamata *activity recognition* per determinare non la posizione del dispositivo ma la modalità di spostamento. Vedremo infatti come capire se l'utente sta camminando, correndo oppure utilizzando un altro mezzo durante i propri spostamenti. Come è facile intuire si tratta di un testo con molti concetti e soprattutto numerosi esempi contestualizzati alla realizzazione dell'applicazione UGHO che, è bene precisare, non rappresenta un prodotto finito ma contiene diversi punti di personalizzazione che possono essere riutilizzati in altri progetti.

Ringraziamenti

La scrittura di un libro può essere paragonata a un viaggio molto lungo. Anche quando si affrontano argomenti consolidati si scopre sempre qualcosa di nuovo che porta entusiasmo e voglia di continuare. Spesso però non mancano le difficoltà legate al fatto che comunque ci sono anche altre cose da fare, le scadenze incombono e non sempre si ha l'ispirazione per descrivere le cose utilizzando un linguaggio il più comprensibile possibile. Le difficoltà sono però quello che rendono questa esperienza degna di essere vissuta. Scrivere un libro equivale a leggerne più di cento. Come tutte le avventure, sono comunque più belle se vissute in compagnia di persone che ti danno un sostegno, che sono poi quelle che devo ringraziare. Prime fra tutte mia moglie Cristina e le mie figlie Alice e Marta che, dopo avermi seguito a Londra, mi hanno sostenuto anche questa volta. Oltre a loro devo ringraziare il mio amico (e collega) Simone Casagrande che, oltre che "stressarmi" piacevolmente con la sua passione per gli algoritmi, mi ha dato suggerimenti sugli argomenti da trattare in questo libro. Infine ringrazio tutti coloro che mi hanno incoraggiato in questa avventura all'estero, che mi ha permesso di rinascere sotto tutti i punti di vista.