



**SAPIENZA**  
UNIVERSITÀ DI ROMA

## **Diario Diabete**

Corso di Tecniche Audiovisive

Alessio Perilli 1321699

A.A. 2014-2015

# Indice

<b>1 Introduzione</b> .....	<b>3</b>
1.1 Motivazione ed obiettivo .....	3
1.2 Organizzazione della tesi .....	3
<b>2 Il diabete</b> .....	<b>4</b>
2.1 Cos'è il diabete? .....	4
2.2 I valori glicemici.....	5
2.3 I tipi di diabete .....	5
<b>3 Android</b> .....	<b>7</b>
3.1 Cos'è Android? .....	7
3.2 La storia di Android .....	7
3.3 L'architettura di Android .....	10
3.4 Ambiente di sviluppo.....	11
3.4.1 Eclipse .....	11
3.4.2 Android studio.....	12
3.5 Applicazione Android .....	12
3.5.1 Android Manifest.....	13
3.5.2 Activity .....	13
3.5.3 View e ViewGroup .....	15
3.5.4 Button o Pulsanti .....	15
<b>4 La mia applicazione: “Diario Diabete”</b> .....	<b>17</b>
4.1 Descrizione dell'applicazione .....	17
4.2 Le classi utilizzate .....	21

# Capitolo 1

---

## Introduzione

### 1.1 Motivazione ed obiettivo

Il mio progetto si basa sulla realizzazione di un'applicazione Android che ha lo scopo di semplificare la routine quotidiana delle persone affette da diabete tipo 1.

Diario Diabete si propone come valida alternativa per sostituire il diario cartaceo con cui tutte le persone affette da questa patologia hanno a che fare ogni giorno.

Essendo diabetico dall'età di 16 anni, ogni qualvolta effettuo il controllo della glicemia e/o mi inietto la dose di insulina, devo riportare il tutto su un diario cartaceo, in modo tale che quando ho la visita di controllo il mio diabetologo può avere una visione di come io riesca a gestire la malattia, se la terapia sta funzionando o meno, e di conseguenza prendere le misure necessarie. Fin qui niente di strano, se non fosse per il fatto che per fare tutto questo nel migliore dei modi, dovrei avere sempre con me il diario e scrivere volta per volta il valore della glicemia e le unità di insulina che sto facendo.

Il problema è proprio questo, portare il diario sempre con se e soprattutto trovare il tempo per aggiornarlo. Nel mio caso, avendo una vita frenetica e mai monotona, mi capita molto spesso di non avere con me il diario, e di doverlo aggiornare solo in un secondo momento, a volte solo poche ore prima della visita.

Da qui la mia idea di implementare su smartphone un vero e proprio diario glicemico (visto che chiunque oggi ha uno smartphone sempre a portata di mano), in modo tale che ogni persona affetta da diabete mellito possa riportare in modo rapido i propri valori di glicemia e le unità di insulina che giornalmente ha eseguito.

### 1.2 Organizzazione della tesi

La tesi è strutturata nel modo seguente. Nel capitolo 2 viene fornita una panoramica della malattia in questione, fornendo vari parametri per la diagnosi del diabete mellito e le diverse tipologie di diabete. Il capitolo 3 descrive il sistema operativo Android, oltre a fornire una panoramica sulla sua architettura e sugli ambienti di sviluppo ad oggi esistenti. Infine nel capitolo 4 viene descritta l'applicazione realizzata.

# Capitolo 2

---

## Il Diabete

### 2.1 Cos'è il diabete?

Diabete è un termine derivato dal verbo greco *διαβαίνειν* che significa “passare attraverso” e identifica alcune malattie caratterizzate da poliuria (abbondante produzione di urina) e polidipsia (abbondante ingestione di acqua).

Il diabete mellito è una forma di diabete, ovvero un gruppo di disturbi metabolici accomunati dal fatto di presentare una persistente instabilità del livello glicemico del sangue, passando da condizioni di iperglicemia a condizioni di ipoglicemia.

Quando l'insulina è prodotta in quantità non sufficiente dal pancreas oppure le cellule dell'organismo non rispondono alla sua presenza, nel sangue si avranno livelli di glucosio più alti del normale (iperglicemia), favorendo la comparsa del diabete mellito. La diagnosi di diabete è certa con un valore di glicemia di 200 mg/dl, rilevato in qualunque momento della giornata o due ore dopo un carico di glucosio.

Il diabete mellito è una malattia cronica caratterizzata dall'aumento della concentrazione di glucosio nel sangue. Responsabile di questa condizione è un difetto nella produzione o nella funzionalità dell'insulina, un ormone secreto a livello del pancreas e indispensabile per il metabolismo degli zuccheri.

Tutti gli zuccheri semplici e complessi (amidi), che vengono assunti con l'alimentazione, sono trasformati nel corso della digestione in glucosio, il quale rappresenta la principale fonte di energia per i muscoli e gli organi. La presenza dell'insulina è di fondamentale importanza, affinché il glucosio possa fare il suo ingresso nelle cellule ed essere utilizzato come carburante.

L'insulina è prodotta da particolari cellule del pancreas (cellule beta) riunite in gruppi chiamati “Isole di Langherans”. Quando l'insulina è prodotta in quantità non sufficiente dal pancreas oppure le cellule dell'organismo non rispondono alla sua presenza, i livelli di glucosio nel sangue tendono ad innalzarsi favorendo la comparsa del diabete.

Molti studiosi individuano nel diabete la malattia del ventesimo secolo. Non a caso la percentuale di popolazione mondiale affetta viene stimata intorno al 5%; circa il 90% della popolazione diabetica è affetta da DM di tipo 2. In Italia la percentuale di individui affetti da tale patologia è mediamente del 3%. L'organizzazione mondiale della sanità (OMS) stima che entro il 2030 più di 360 milioni di persone in tutto il mondo saranno affette da diabete.

## 2.2 I valori glicemici

La concentrazione di glucosio nel sangue si misura con la glicemia. In soggetti sani, che hanno una vita regolare e un'alimentazione corretta, generalmente nell'arco della giornata i valori della glicemia si mantengono tra i 60 e i 130 mg/dl. A digiuno, i valori glicemici possono variare dai 70 ai 110 mg/dl; tra 110 e 125 mg/dl si tratta di condizione di alterata glicemia a digiuno, una condizione che dovrebbe invitare il paziente a porre maggior attenzione al suo stile di vita e in particolare alla sua alimentazione.

Valori di glicemia uguali o superiori a 126 mg/dl, secondo l'American Diabetes Association, sono da considerarsi probabili sintomi di diabete.

La diagnosi di diabete è certa con un valore di glicemia di 200 mg/dl, rilevato in qualunque momento della giornata o due ore dopo un carico di glucosio. Valori di glicemia compresi fra 140 a 200 mg/dl dopo un carico di glucosio definiscono, invece, la ridotta tolleranza al glucosio.

## 2.3 I tipi di diabete

Il diabete mellito è una forma di diabete, ovvero un gruppo di disturbi metabolici accomunati dal fatto di presentare una persistente instabilità del livello glicemico del sangue, passando da condizioni di iperglicemia a condizioni di ipoglicemia.

Attualmente si distinguono tre diverse forme di diabete:

- 1. il diabete di Tipo 1:** un tempo chiamato diabete insulino-dipendente o diabete giovanile, riguarda il 10% dei casi di diabete e si sviluppa prevalentemente a partire dall'infanzia e dall'adolescenza. Nel diabete di tipo 1, la produzione di insulina da parte del pancreas viene soppressa o fortemente ridotta a causa della distruzione delle cellule beta da parte del sistema immunitario. *Le cause di questa malattia sono ancora sconosciute* ma, attualmente, il diabete di tipo 1 è classificato come una *“malattia autoimmune”*, cioè legata a una reazione del sistema immunitario contro l'organismo stesso, scatenata da una concomitanza di fattori genetici e ambientali. Alcuni dei suoi sintomi sono le *“urine abbondanti e frequenti, sete e fame eccessiva, dimagrimento”*;
- 2. il diabete di Tipo 2:** rappresenta la forma di diabete più comune e interessa il 90% dei casi. Prevalentemente, si sviluppa a partire dai 40 anni e *colpisce principalmente i soggetti obesi o sovrappeso*. Nel diabete di tipo 2, il pancreas è in grado di produrre insulina (seppur in maniera ridotta) ma le cellule dell'organismo non riescono a utilizzarla in modo efficiente: ciò comporta un aumento dei livelli di glucosio nel

sangue. In genere, *la presenza di diabete tipo 2 può non essere rilevata per molti anni, in quanto l'iperglicemia si sviluppa gradualmente e non comporta sintomi particolarmente evidenti come quelli del diabete di tipo 1. Alcuni dei sintomi tipici del diabete tipo 2 sono: “sensazione di stanchezza, frequente bisogno di urinare anche nelle ore notturne, sete inusuale, perdita di peso, visione offuscata e lenta guarigione dalle ferite”;*

3. **il diabete gestionale:** per diabete gestionale si intende un aumento dei livelli di glucosio che si manifesta o viene rivelato per la prima volta nel periodo della gravidanza. Questa condizione si verifica nel 4% delle donne in stato interessante. Generalmente, il diabete gestionale tende a scomparire al termine della gravidanza, tuttavia, le donne che ne hanno sofferto presentano un rischio più elevato di sviluppare diabete di tipo 2 in età avanzata.

# Capitolo 3

---

## Android

### 3.1 Cos'è Android?

Android è il sistema operativo sviluppato da Google per dispositivi mobili. È stato progettato principalmente per smartphone e tablet, con interfacce utente specializzate per televisori (Android Tv), automobili (Android Auto), orologi da polso (Android Wear), occhiali (Google Glass), ed altri.



Fig. 1: Google Android mascot.

Il colosso statunitense, con l'avvento di questa piattaforma, ha messo a disposizione dell'intera popolazione globale un proprio insieme di librerie e strumenti volti allo sviluppo e alla creazione di applicazioni mobili.

### 3.2 La storia di Android

La società Android Inc. è stata fondata nel ottobre del 2003 a Palo Alto da Andy Rubin, Rich Miner, Nick Sears e Chris White. Il loro scopo era quello di progettare software per dispositivi mobili.

Il 15 agosto del 2005 Google acquista Android e con essa anche il suo team di sviluppatori, con lo scopo di realizzare un sistema operativo mobile basato sul kernel Linux.

Nel novembre del 2007 in occasione della nascita dell'organizzazione OHA (consorzio formato da Google, da produttori di device quali Motorola e Samsung e da operatori mobili come Vodafone e T-Mobile), l'alleanza decise di rilasciare il primo software development kit (SDK) e il lancio della versione 1.0 Apple Pie.

Questa prima versione comprendeva l'utilizzo del market per scaricare le apps, il browser, il supporto delle reti Wi-fi, l'accesso ai servizi di posta elettronica e alle applicazioni Google. Avendo a disposizione solamente un emulatore, i vari sviluppatori di apps dovettero attendere circa un anno prima di vedere realmente implementate le loro creazioni su un dispositivo mobile. Per l'appunto il 23 settembre 2008 nasce G1 di T-Mobile, il primo smartphone con sistema operativo Android.

Da qui in poi si susseguono molteplici aggiornamenti e sviluppi delle versioni Android:

- 1.1 Petite Four (09-02-2009), update per risolvere vari bug.
- 1.5 Cupcake (30-04-2009), per una maggiore integrazione dei servizi Google, oltre a supportare i widget.
- 1.6 Donut (15-09-2009), che aggiunge la ricerca vocale e testuale per i contenuti presenti in locale e sul Web. Introduce la sintesi vocale e la gesture, oltre ad assicurare il supporto 3G CDMA.
- 2.0 Eclair (26-10-2009), che introduce la novità di inviare dati tramite l'ausilio della tecnologia Bluetooth, oltre all'inserimento di numerose funzionalità per la fotocamera e all'aggiunta del supporto per il multi-touch e i live wallpaper. Ulteriori miglioramenti si sono avuti nelle versioni Eclair 2.0.1 ed Eclair 2.1 .
- 2.2 Froyo (20-05-2010), presentata in occasione della Google I/O conferenced 2010, introduce le funzioni di tethering Wifi per utilizzare il terminale come hot spot wireless per connettersi a internet, Adobe Flash player 10.1, Adobe AIR, oltre alla possibilità di salvare le apps direttamente sulla memoria. In seguito sono state sviluppate le versioni Froyo 2.2.1, Froyo 2.2.2 e Froyo 2.2.3 .
- 2.3 Gingerbread (06-12-2010), è la versione più diffusa sui terminali Android e la prima a supportare le chat video e vocali attraverso Google Talk. Comprende inoltre il supporto alla tecnologia NFC e un aggiornamento per la tastiera. Nello stesso anno vennero implementate le versioni di Gingerbread 2.3.3 , 2.3.4, 2.3.5, 2.3.6 e per finire la versione 2.3.7 .
- 3.0 Honeycomb (22-02-2011), versione dedicate solo ai tablet. Nell'arco di circa 5 mesi, da maggio ad agosto, si sono susseguite le versioni 3.1 , 3.2 , 3.2.1 e 3.2.2 .
- 4.0.1 Ice Cream Sandwich (19-10-2011), ha la particolarità di essere la prima release della piattaforma pensata sia su smartphone che su tablet. Consente un sostanziale incremento delle prestazioni oltre ad avere pulsanti virtuali al posto di quelli hardware. Studi successivi hanno portato alla creazione degli aggiornamenti 4.0.2, 4.0.3 e 4.0.4 .



- 4.1.1 Jelly Bean (27-06-2012), che consente un sostanziale miglioramento del riconoscimento del tocco e del riconoscimento vocale, l'ottimizzazione dell'utilizzo della CPU, miglioramenti della fluidità oltre all'aggiunta di importanti ottimizzazioni per la fotocamera. In questa versione si abbandona l'utilizzo di Adobe. Si è passati poi alle release 4.1.2 , 4.2, 4.2.1 e 4.2.2. Un anno dopo Google rilascia 4.3 Jelly Bean che porta diverse migliorie, tra cui il supporto all'OpenGL ES 3.0 e il Bluetooth 4.0.
- 4.4 KitKat (31-10-2013), Google inizia la distribuzione per dispositivi Nexus ad esclusione del Galaxy Nexus. In poco tempo Android KitKat gira sul 21% di tutti i dispositivi Android. Seguono le versioni 4.4.1, 4.4.2, 4.4.3 e 4.4.4 .
- 5.0 Lollipop (03-09-2014) , si verifica una netta svolta per quel che riguarda tutta l'interfaccia del sistema attraverso la completa ridisegnazione delle icone in uno stile più semplice e piatto, abbandonando quindi lo stile Holo che fino a quel momento aveva caratterizzato Android. In concomitanza alla sua uscita vengono presentati i modelli HTC Nexus 9, Asus Nexus Player e Motorola Nexus 6. Nei mesi successivi si sono susseguite le release 5.0.1, 5.0.2, 5.1 e 5.1.1 , che hanno portato vari miglioramenti e alla risoluzione di vari bug.
- Il 17 agosto 2015 nel giardino del Googleplex , viene presentata la statua e il nome ufficiale di Android M, che si chiamerà Android Marshmallow



Fig. 2: Statua di Android Marshmallow.

### 3.3 L'architettura di Android

Android ha un'architettura molto complessa. La sua piattaforma è stata progettata così bene al punto da essere alla pari dei sistemi progettati esclusivamente per desktop o laptop.

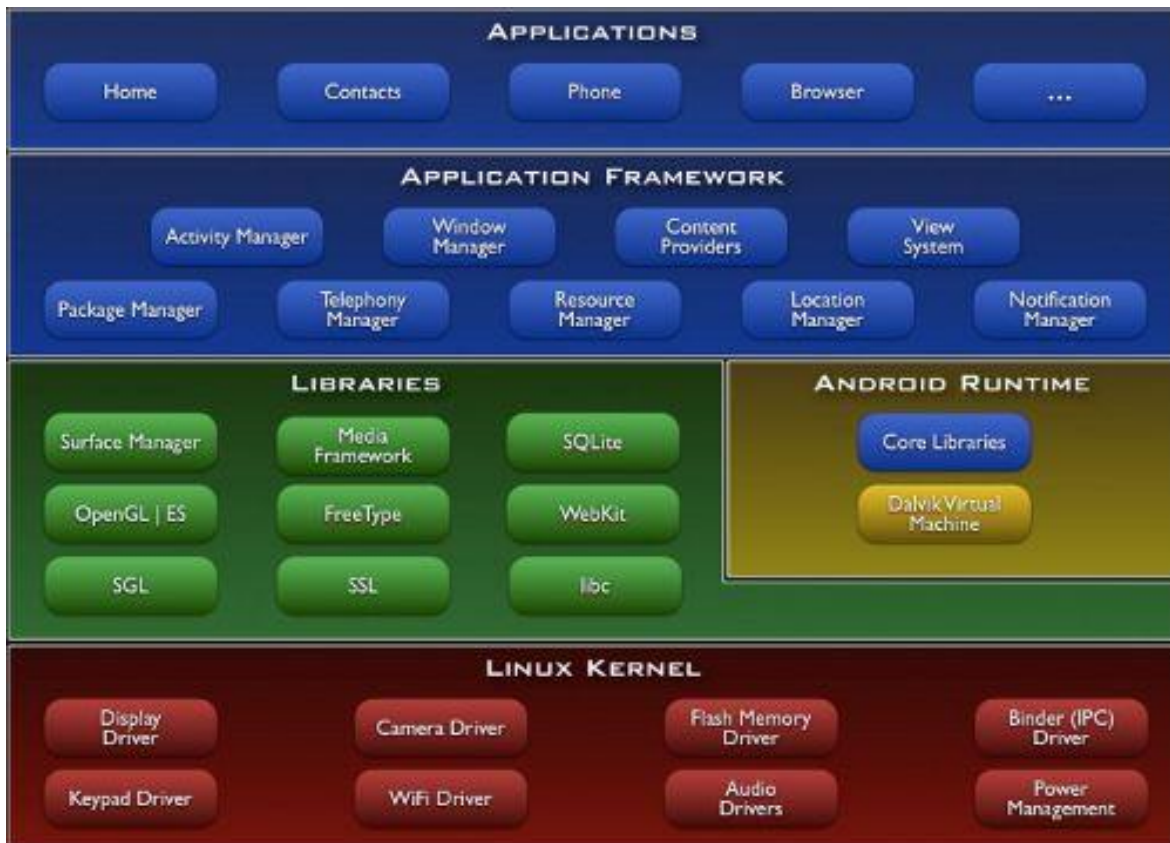


Fig. 3 : Architettura di Google Android.

Come si evince dalla fig.3, Google ha attinto a piene mani dal mondo Open Source. Il cuore del sistema è basato sul kernel di Linux (versione 2.6), il quale include i vari driver per la gestione hardware del dispositivo, ovvero i driver per il Wifi, per il Gps, per il bluetooth, per la tastiera, per lo schermo, per il controllo dell'audio, per la fotocamera e per il touchscreen. I vari produttori di telefoni possono intervenire già a questo livello per modificare i vari driver in modo che siano compatibili con i propri smartphone.

Sopra il kernel vi sono le librerie fondamentali della piattaforma Android, anche queste derivate dal mondo Open Source. Da menzionare la libreria OpenGL ES, che gestisce la grafica 2D e 3D; SQLite, per la gestione dei dati; WebKit, che garantisce la visualizzazione delle pagine web.

L'ambiente Runtime viene gestito da una libreria Core e da una macchina virtuale, la Dalvik Virtual Machine (DVM); insieme danno vita alla vera e propria piattaforma di sviluppo per Android. In sostanza la macchina virtuale è una versione della Java Virtual Machine, progettata per girare su

smartphone. Con gli ultimi sviluppi tecnologici, e con l'avvento di telefoni dual core o addirittura quad core, questa macchina virtuale è diventata ancor più performante rispetto ai modelli iniziali. Il penultimo livello è occupato dai gestori delle applicazioni e di sistema, mentre sullo strato più alto dell'architettura troviamo le applicazioni utente, alcune delle quali sono già incluse nel pacchetto di base, come Home, Rubrica e Browser. È proprio a questo livello che si inseriscono le applicazioni che lo sviluppatore intende realizzare per il dispositivo.

### 3.4 Ambiente di sviluppo

Per poter creare applicazioni che riescano a girare su sistemi Android è di fondamentale importanza installare sul proprio pc un apposito kit di sviluppo (Android SDK), che fornisce le librerie API e gli strumenti necessari alla creazione, alla prova e al debug delle app che ogni programmatore avrà intenzione di implementare. Nel seguito verranno illustrati i due principali ambienti di sviluppo utilizzati dai programmatori.

#### 3.4.1 Eclipse

Eclipse è l'ambiente di sviluppo integrato multi-linguaggio e multipiattaforma utilizzato nella produzione di svariati software; si passa da un completo IDE<sup>1</sup> per Java ad un ambiente di sviluppo per il linguaggio C++, oltre alla capacità di poter gestire XML<sup>2</sup> e PHP<sup>3</sup>.

L'utilizzo di Eclipse come ambiente di sviluppo delle applicazioni Android è evidenziato dal fatto che la singola applicazione può essere scomposta in una parte dinamica, scritta in Java; e una parte statica, scritta in XML.

Android non è un linguaggio nuovo vero e proprio, ma una forma di Java. Lo stesso vale per la macchina virtuale, visto che Android non usa la Java virtual machine(JVM), ma ne ha a disposizione una propria, la Dalvik virtual machine(DVM), come già accennato nel paragrafo precedente.

Mentre la JVM esegue bytecode, la DVM non esegue bytecode standard ma uno specifico tipo di linguaggio chiamato DEX (Dalvik EXecutable), il quale è stato realizzato appositamente per garantire un migliore rendimento su smartphone.

La sostanziale differenza tra il linguaggi Java e Android sta nel fatto che, mentre Java ha un programma principale o metodo main da cui inizia a caricare il software e ad avviarsi; l'applicazione Android è creata per essere pilotata dagli eventi dell'hardware o dei vari componenti.

---

<sup>1</sup>Integrated Development Environment, o ambiente di sviluppo integrato; è il software che facilita il lavoro dei programmatori in fase di programmazione.

<sup>2</sup>eXtensible Markup Language, o linguaggio di markup[25].

<sup>3</sup>Hypertext Preprocessor[26].

Il vantaggio sostanziale di questo tipo di implementazione sta nell'ottimizzazione del sistema operativo, il quale evita di caricare componenti non supportati e sottoutilizzati.

### 3.4.2 Android Studio

Android Studio è l'IDE (Integrated Development Environment) ufficiale di Android, ovvero l'ambiente di sviluppo integrato per tutti gli sviluppatori Android, basato su IntelliJ IDEA. Fornisce nuove funzionalità e miglioramenti rispetto a Eclipse ADT. Sviluppare con questa nuova piattaforma è più semplice e funzionale: *“addio workspace, benvenuti modules e library”*. Presto sarà l'unico ambiente di sviluppo supportato da Google.

Android Studio offre:

- un sistema di compilazione flessibile basato su Gradle;
- possibilità di costruire APK varianti e multipli;
- un esteso supporto ai servizi Google e a vari tipi di dispositivi;
- ricco ambiente di layout con editing a tema;
- strumenti per la cattura di prestazioni, usabilità, compatibilità di versione, e altri problemi;
- funzionalità ProGuard e app-signing;
- include il supporto per Google Cloud Platform, e rende più semplice integrare Google Cloud Messaging e App Engine.

### 3.5 Applicazione Android

Le applicazioni definiscono le parti software che possono essere installate su Android. Tali apps possono essere scaricate direttamente dal market offerto da Google, oppure sviluppate direttamente su pc e poi installate sullo smartphone. Per la realizzazione dell'applicazione, si è deciso di lavorare con Android, al contrario di iOS o WindowsPhone, principalmente per il fatto che Android è un sistema Open Source, ma anche perché meno oneroso dal punto di vista economico delle altre piattaforme, in quanto richiedono il pagamento di una licenza annuale per poter creare apps e poterle inserire nei rispettivi store. Il terzo vantaggio risiede nel fatto che Android è molto più flessibile e permette di apportare modifiche al proprio dispositivo, cosa che lo rende molto più appetibile delle altre piattaforme per la stragrande maggioranza degli sviluppatori di app.

### 3.5.1 Android Manifest

Ogni applicazione Android deve avere un file denominato `AndroidManifest.xml` nella sua directory root, che viene generato automaticamente all'atto della creazione del progetto. Il manifesto presenta tutte le informazioni essenziali a riguardo dell'applicazione che si sta creando, oltre a fornire le corrette istruzioni per l'esecuzione dell'applicazione stessa. Inoltre, il manifesto:

- identifica univocamente il package di Java per l'applicazione;
- descrive tutte le componenti dell'applicazione ovvero le attività, i servizi e i content provider di cui l'app è composta; inoltre nomina le classi che implementano ogni componente e ne rende pubblica la loro capacità;
- determina i processi che ospiteranno i componenti dell'applicazione;
- dichiara quali permessi devono avere le applicazioni per poter accedere alle parti protette delle API e poter interagire con altre applicazioni;
- dichiara quali permessi devono avere le altre applicazioni per poter interagire con i vari componenti;
- dichiara il minimo livello di Android API che l'applicazione richiede;
- elenca le librerie alle quali l'applicazione deve essere collegata.

### 3.5.2 Activity

Si classifica sotto il nome di attività la capacità dell'utente di fare qualcosa, come ad esempio scrivere un messaggio, comporre un numero telefonico o scattare una foto. Per avviare a queste azioni, l'utente deve poter interagire con il dispositivo. Tra i vari mezzi di interazione con lo smartphone vi sono la tastiera e il touch-screen, ma affinché l'utente sappia cosa fare o il software possa visualizzare i risultati dell'elaborazione, un ruolo fondamentale è svolto dal display.

Dal punto di vista del programmatore, l'attività è definita come una classe che estende `android.app.Activity`, visto che nella creazione del codice lo sviluppatore utilizza i metodi ereditati dalla classe `Activity` per interagire con l'utente, intercettare i cambi di stato e controllare cosa accade sul display. Le applicazioni Android hanno carattere di esclusività, e anche se è possibile mandare in esecuzione più di un'attività simultaneamente, solo una di queste può effettivamente occupare il display, che a differenza dei pc, negli smartphone ha dimensioni molto ridotte. Quindi si ha che mentre l'attività visualizzata sul display è in esecuzione, le altre attività sono ibernata, con la possibilità di essere recuperate e ripristinate in seguito, secondo le politiche definite dal ciclo delle attività.

I metodi di callback implementati nel ciclo delle attività sono:

- onCreate();
- onStart();
- onResume();
- onPause();
- onStop();
- onDestroy();
- onRestart().

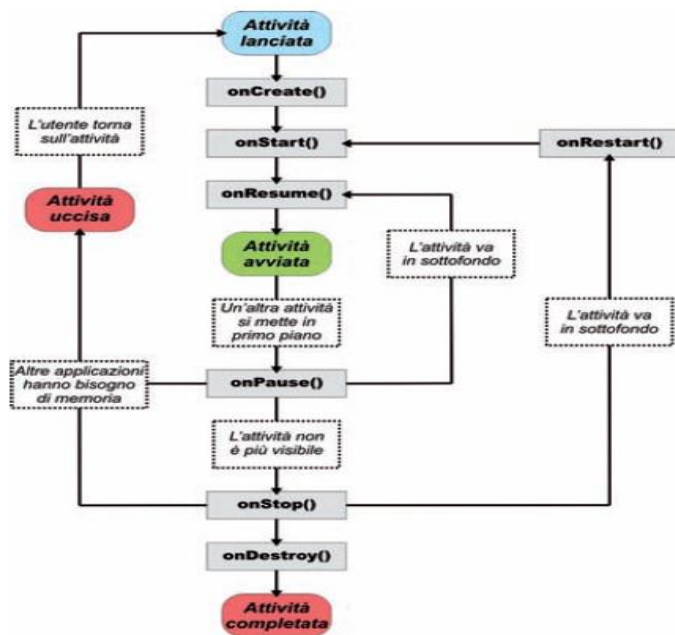


Fig. 4: ciclo di vita di un'attività

Di seguito viene riportata la tabella in cui vengono descritti tutti i metodi di callback che compongono il ciclo di vita dell'attività.

Metodo	Descrizione
<b>onCreate()</b>	Questo metodo viene richiamato ogni volta che un'attività viene creata. È seguito sempre dal metodo onStart().
<b>onRestart()</b>	Questo metodo viene chiamato dopo che l'attività è stata interrotta e appena prima che questa venga ripresa in esecuzione. È seguito sempre dal metodo onStart().
<b>onStart()</b>	Questo metodo viene richiamato un attimo prima che l'attività venga resa visibile all'utente. A seconda delle circostanze, può essere seguito dal metodo onResume() o dal metodo onStop().
<b>onResume()</b>	Questo metodo viene richiamato un attimo prima che l'attività inizia ad interagire con l'utente. È seguito sempre dal metodo onPause().
<b>onPause()</b>	Questo metodo viene richiamato quando si è in procinto di iniziare a riprendere un'altra attività. Può essere seguito dal metodo onResume() o dal metodo onStop().
<b>onStop()</b>	Questo metodo viene richiamato quando l'attività non è più visibile all'utente. Seguono il metodo onRestart() o il metodo onDestroy().
<b>onDestroy()</b>	Questo metodo è chiamato prima che l'attività venga distrutta, in particolare è l'ultima chiamata che l'attività riceverà

Tab. 1: Eventi che caratterizzano il ciclo di vita dell'attività.

### 3.5.3 View e ViewGroup

Si identifica col nome di View l'elemento base per i componenti dell'interfaccia utente. In particolare, una View corrisponde al modo con cui Android gestisce tutto ciò che è presente sullo schermo. Tipici oggetti View sono i bottoni, le icone, i campi di testo, e tutti gli altri componenti dell'interfaccia grafica. Quindi da qui si deduce che la View è la classe base per i widget, utilizzati per creare tutte quelle componenti grafiche sopra citate.

Tutti i componenti View estendono la classe android.view.View.

Si definisce la sottoclasse ViewGroup come la classe base per i layout<sup>4</sup>, che sono contenitori invisibili che, essendo a loro volta oggetti View, contengono altri componenti ViewGroup, andando a definire le loro proprietà di layout.

Il componente ViewGroup estende la classe android.view.ViewGroup.

In questo modo è possibile organizzare le componenti dello schermo secondo un'organizzazione ad albero.

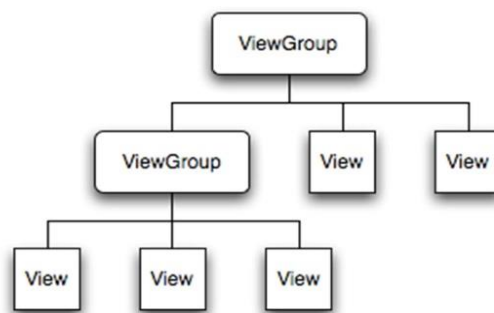


Fig. 5: Organizzazione dei componenti di base dello schermo secondo View e ViewGroup.

### 3.5.4 Button o Pulsanti

Il pulsante è uno dei componenti più classici della piattaforma Android, visto che è utilizzato per dare il via alle operazioni. In particolare, è costituito da un testo o icona che specifica l'azione che si verifica quando l'utente lo tocca. A seconda del tipo di pulsante che si vuole creare nel layout, si hanno a disposizione tre modi diversi:

1. pulsante con solo testo, usando la classe Button;

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/button_text"
    ... />
```

Fig. 6: Codice per creare pulsante con testo al suo interno.

<sup>4</sup>Disposizione o impaginazione; comprende tutte i ViewGroup utilizzati per posizionare widget sullo schermo.

2. pulsante con un'icona, usando la classe ImageButton;

```
<ImageButton  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:src="@drawable/button_icon"  
    ... />
```

Fig. 7: Codice per creare pulsante con icona

3. pulsante con icona e testo, usando la classe Button con l'attributo android:DrawableLeft.

```
<Button  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="@string/button_text"  
    android:drawableLeft="@drawable/button_icon"  
    ... />
```

Fig. 8 : Codice per creare pulsante con icona e testo.



# Capitolo 4

---

## La mia applicazione: “Diario Diabete”

### 4.1 Descrizione dell'applicazione

Diario Diabete è un applicazione Android per smartphone che permette di emulare il diario glicemico, in modo tale da poter memorizzare con estrema facilità i valori di glicemia e insulina. Il mio obiettivo principale è stato quello di realizzare un'app che fosse prima di tutto semplice da utilizzare, visto che il diabete mellito tipo 1 è una patologia che colpisce principalmente i bambini; e in secondo luogo che riesca a sensibilizzare quelle persone che pur affette da diabete, evitano di utilizzare il diario, il quale risulta essere uno strumento fondamentale per valutare se la terapia a cui si è sottoposti sta funzionando o meno. Per quanto riguarda la scelta dell'icona dell'app, ho utilizzato il simbolo universale del diabete:



Fig. 9: Simbolo della giornata mondiale del diabete

Una volta installata l'app sullo smartphone, al suo primo avvio si presenta la seguente schermata:

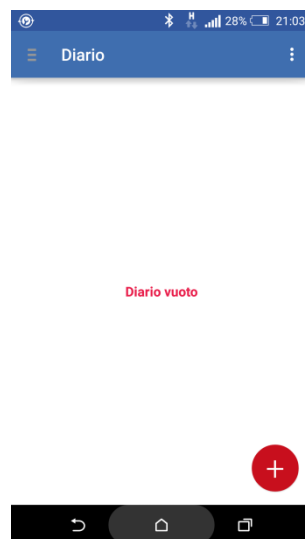


Fig. 10: schermata all'avvio dell'app.

Andando a cliccare il pulsante in alto a sinistra si può andare in qualsiasi fragment dell'app:

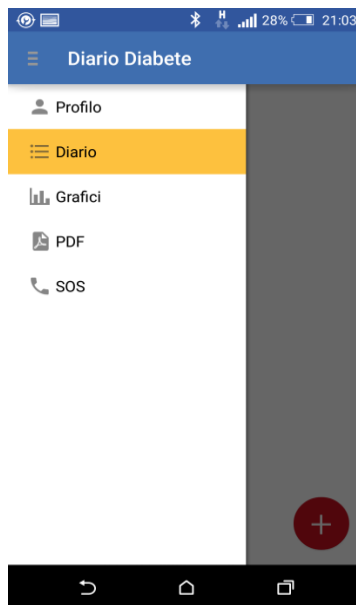


Fig. 11: Menù.

1. Profilo, attraverso il quale vengono inseriti i dati personali dell'utilizzatore;
2. Diario, permette di tornare alla schermata iniziale dove vengono visualizzate le misurazioni inserite;
3. Grafici, il quale visualizza un grafico giornaliero dell'andamento della glicemia e allo stesso tempo un grafico in cui viene riportato il valore medio glicemico per ogni giorno del mese;
4. PDF, permette di realizzare un file pdf, il quale riporta tutti i valori che sono stati inseriti nel diario, con relativa data e ora;
5. SOS, ci permette di fare una chiamata di emergenza ad un numero di fiducia che viene richiesto nella fase di aggiornamento del profilo.

Di seguito sono riportati gli screenshot dei fragment di cui si compone Diario Diabete.

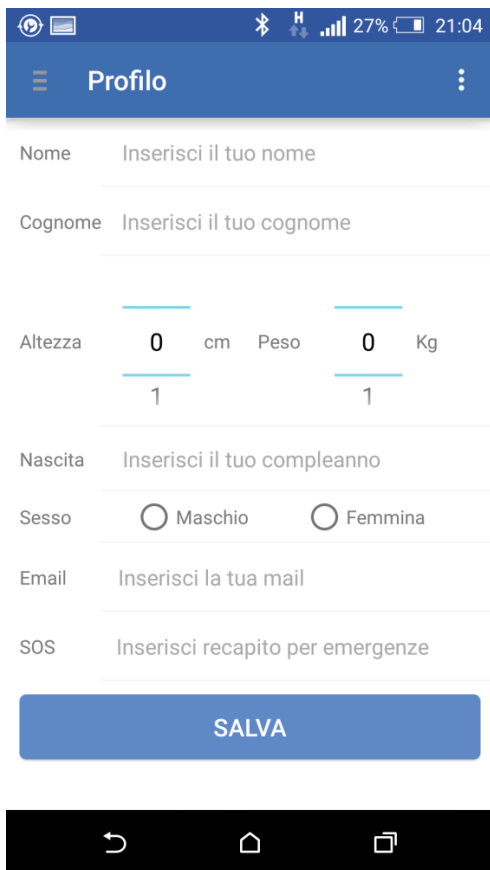


Fig. 12: Profilo.

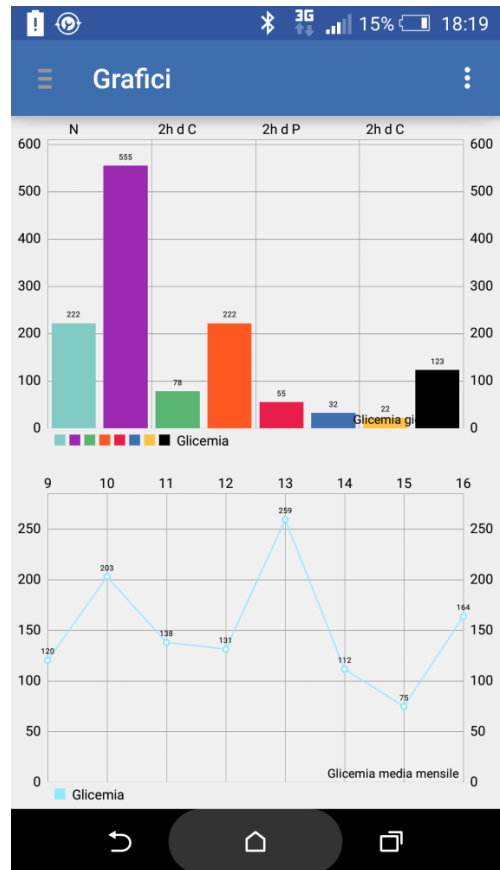


Fig. 13: Grafici.

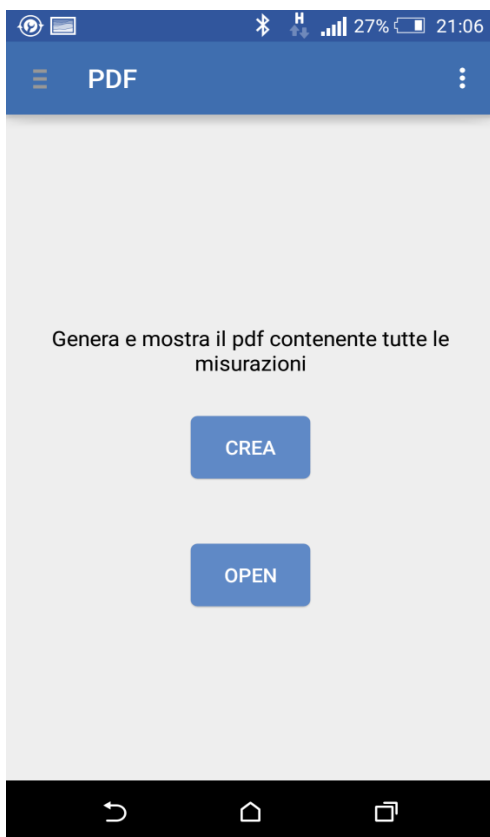


Fig. 14: PDF.

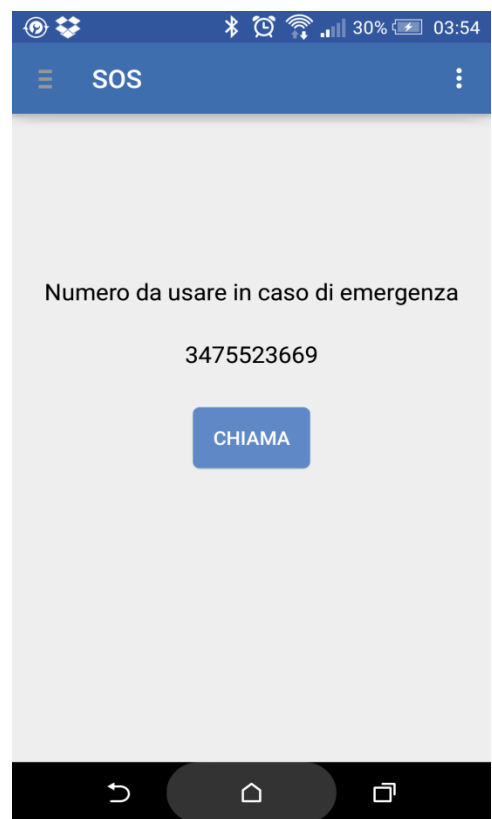


Fig. 15: SOS.

Cliccando in alto a destra si accede al menu che riporta la legenda dei colori utilizzati nei grafici, oltre ad un bottone utilizzato per effettuare la chiamata di emergenza in modo rapido.

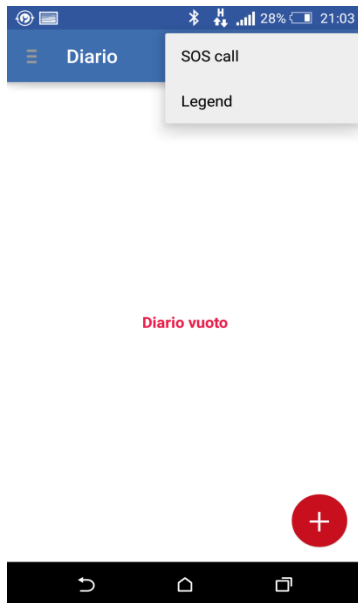


Fig. 16: Screenshot del menu.



Fig. 17: Legenda periodo giornata

Infine, vi è il cuore di Diario Diabete, ovvero la fase di inserimento dei dati. Come prima cosa si clicca sul pulsante rosso e di seguito viene visualizzato il seguente dialog con cui poter inserire il valore della glicemia, le unità di insulina iniettate (opzionale), il periodo della giornata in cui viene inserita la misurazione, la data e l'ora.

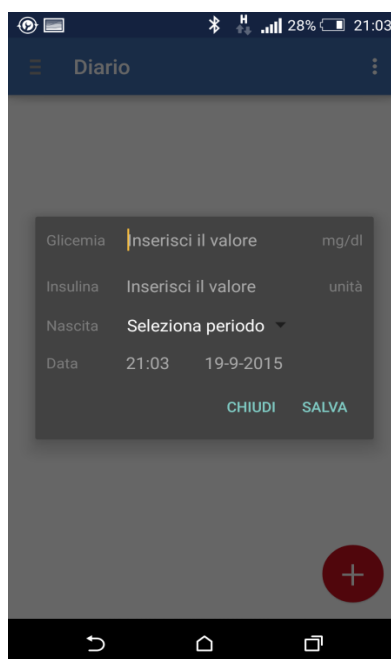


Fig.18: Fase di inserimento.

## 4.2 Le classi utilizzate

Per la realizzazione di Diario Diabete sono state utilizzate le seguenti classi :

- **Fragments:** i fragment sono uno dei più importanti elementi per la creazione di un'interfaccia utente Android moderna. Un fragment è una porzione di activity; non si tratta solo di un gruppo di controlli o di una sezione di layout, ma può essere definito come una specie di sub-activity con un suo ruolo funzionale molto importante ed un suo ciclo di vita.

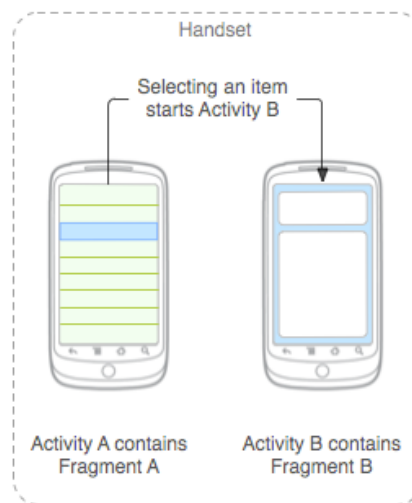


Fig. 19: Esempio di moduli definiti da fragment combinati in un'attività.

- **ListView e ArrayAdapter:** le ListView sono un particolare widget per la visualizzazione di elementi in una lista scorrevole, mentre le voci appartenenti al componente sono caricate in automatico da un Adapter che si occupa sia dell'accesso ai dati sia della visualizzazione.
- **DialogFragment:** è un frammento che visualizza una finestra di dialogo, galleggia sulla parte superiore della finestra della sua attività. Questo frammento contiene un oggetto di dialogo, che viene visualizzato a seconda dei casi in base allo stato del frammento.

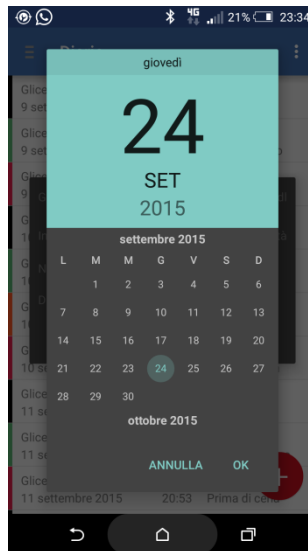


Fig. 20: DateDialog di Diario Diabete

- **SQLiteOpenHelper:** è bene implementare una classe helper per semplificare le interazioni con il database ed introdurre così un livello di astrazione che fornisca metodi intuitivi, flessibili e robusti per inserire, eliminare e modificare i record del database. La classe helper è il luogo ideale per memorizzare le costanti statiche come il nome della tabella e delle colonne.