

# Relazione Programmazione di Sistemi Embedded

## Progetto da 6CFU – Location App

I test sono stati eseguiti su un Samsung S21, aggiornato all'ultima versione di android 12 (SDK version 31).

### Note per il Debug:

- Se si vuole testare l'app usando un emulatore, bisogna usare un dispositivo che supporti i Google Play Services, altrimenti Google Maps e di conseguenza l'app in generale (impedito da Android) NON potrà funzionare.
- A seconda dell'hardware utilizzato, l'intervallo con cui la posizione viene ottenuta potrebbe non essere lo stesso: con un S21, S21FE e S21 Plus usando come intervallo 5 secondi la posizione è ottenuta ad intervalli perfettamente costanti.

Segnalo tuttavia che usando l'emulatore di un Pixel 4 sembra che gli intervalli non siano sempre regolari. Non so se ciò sia legato alla limitata RAM presente nel mio PC che non è in grado di gestire l'emulatore o altro.

### Funzionamento:

L'applicazione è stata realizzata usando 1 activity che ospita 2 fragment:

- 1) Mostra i dati di latitudine, longitudine e altitudine sia in forma testuale che grafica su una mappa
- 2) Mostra il percorso compiuto negli ultimi 5 minuti su una mappa. Non essendo possibile visualizzare l'altitudine su una mappa 2d, si è optato per l'inserimento di un grafico per mostrare tale informazione.  
I valori in tale interfaccia vengono *aggiornati in tempo reale*, ogni volta che una nuova posizione viene recuperata.

Le mappe utilizzate sono quelle del framework di Google Maps, mentre per il grafico dell'altitudine sopra citato si è utilizzato il framework MPAndroidChart.

Per il recupero della posizione si è deciso di optare per fusedLocationProviderClient invece di LocationManager in quanto questo permette sempre il reperimento di altitudine, longitudine e latitudine in modo a mio avviso più agevole (con LocationManager il dato relativo all'altitudine era rilevato solo con l'utilizzo di GPS\_PROVIDER e non NETWORK\_PROVIDER).

Il tempo tra una richiesta e l'altra è stato impostato a 5 secondi.

All'avvio dell'applicazione vengono richiesti i permessi di localizzazione (sia COARSE che FINE, come richiesto dalle ultime versioni di android) e di attivare il GPS (qualora queste operazioni non siano già state fatte).

La connessione di MainActivity al Service avviene mediante binding, così da avere un riferimento ai valori presenti nel Service. Siccome si vuole gestire il ciclo di vita separatamente rispetto al ciclo di vita dell'activity, il Service funziona anche in modalità started (Service quindi in modalità Started + Bound).

Siccome si vuole reperire la posizione in background, è necessario che il Service sia un Foreground Service.

Quando un nuovo oggetto viene recuperato, i fragment saranno informati di ciò mediante l'utilizzo di un LocalBroadcastReceiver e potranno così aggiornare la loro interfaccia. I dati verranno recuperati dal riferimento al service che i fragment hanno restituito dal metodo onBind.

La collezione dei dati termina quando l'applicazione viene chiusa (eliminata dalle app recenti) da parte dell'utente.

Si sarebbe potuto lasciare l'applicazione sempre in esecuzione, ma si è ritenuto che dover mandare l'utente nelle impostazioni per disabilitare manualmente l'app dalle app in background fosse troppo complicato per l'utente stesso e avrebbe potuto rovinare l'esperienza rendendo l'utente spaesato e confuso davanti ad un'app che non si chiude quando viene rimossa dalle app recenti.

### Fragment 1:

- Card: si aggiornano ogni 5 secondi.
- Mappa: mostra un marker nella posizione in cui il dispositivo si trova. Ogni volta che un nuovo update viene ottenuto, si ricentra la mappa in modo che l'utente possa seguire i suoi movimenti senza dover toccare la mappa. Tuttavia, viene comunque fornita all'utente la capacità di spostare la mappa per vedere dove si trova e cosa c'è nelle vicinanze. Può ritornare a seguire i suoi movimenti automaticamente mediante l'utilizzo del pulsante arancione "CENTER" in alto a destra della mappa. Cliccando sul marker è possibile vedere la propria posizione. Se la posizione viene disabilitata, il marker della posizione corrente viene disattivato.

## Fragment 2:

Si è deciso di fornire all'utente la capacità di vedere l'aggiornamento dei propri spostamenti in tempo reale anziché dover ricaricare ogni volta la pagina per vedere l'andamento degli stessi nel tempo.

I dati vengono aggiornati ogni volta che una nuova posizione viene recuperata, pertanto se la posizione non è abilitata, l'interfaccia rimarrà statica e si aggiornerà solo quando il GPS verrà avviato nuovamente dall'utente.

- Mappa: si mostra sempre con un marker la posizione corrente dell'utente e il percorso fatto per arrivare in tale posizione (senza un marker, non sarebbe stato chiaro quale fosse il punto iniziale e quale quello finale). In questo caso il marker viene mantenuto anche in assenza della posizione per il motivo appena spiegato.
- Grafico dell'altitudine: si è usata la libreria MPAndroidChart. Ogni volta che un nuovo sample è disponibile, tale grafico viene aggiornato con tutti i valori. Sia l'asse x che l'asse y vengono dinamicamente scalati.

## Problematiche o osservazioni varie:

- Dal momento che FusedLocationProviderClient va ad inserire le proprie richieste nel mainLooper (quindi usando il main Thread), questo talvolta può generare problemi di concorrenza con il caricamento della mappa (cosa che avviene sempre nel main thread). Inserendo dei ritardi in fase di avvio, si è tentato di arginare parzialmente tale problema. Raramente, però, il reperimento della posizione può avvenire comunque qualche secondo dopo rispetto all'intervallo definito.

Tale difficoltà è presente **solo** in fase di avvio dell'applicazione. Una volta raggiunta la condizione di regime, l'ottenimento della posizione avviene sempre ogni 5 secondi senza problemi.

Quando bisogna zoommare la mappa alla posizione corrente, sul log di debug avviene un elevatissimo numero di stampe del seguente messaggio:

***I/Counters: exceeded sample count in FrameTime***

che porta via diversi secondi.

Su internet, dopo un'approfondita ricerca, non si è trovata la causa di tale messaggio.

A volte il messaggio di recupero della posizione viene postato nel looper prima dei messaggi sopra citati, a volte dopo che tutti sono finiti (dopo circa 1

secondo). Vedere il file **debug.pdf** presente nel repository per una spiegazione dettagliata analizzando direttamente il log di debug. Usando il metodo `moveCamera()` al posto di `animateCamera()`, l'effetto grafico risulta meno gradevole alla vista (si sposta la telecamera senza animazione) ma permette di ridurre il numero di messaggi che vengono stampati.

- Nel log di debug, in fase di caricamento delle mappe, vengono stampati anche messaggi di errore. Non essendo però degli errori che mandano in crash il programma, si suppone non siano qualcosa di grave. Probabilmente tali messaggi sono legati a Google Ads, ma le informazioni presenti in internet non sono sufficienti per stabilirlo con certezza.

## Lint messages/suppressions:

- **Obsolete SDK\_INT Version Check:** A volte sono stati lasciati dei controlli non necessari - per come è stata impostata la versione minima di Android nel file `.gradle` - deliberatamente per i fini educativi del progetto.
- **Icon colors do not follow the recommended visual style inspection:** onestamente non capisco perchè venga segnalato questo errore in quanto l'immagine usata come icona della top bar è bianca (come richiesto). E' stata colorata manualmente quindi non è chiaro il motivo di tale segnalazione.
- **Proofreading:** Sono dei typo legati a qualche parola scritta in italiano anziché in inglese (il progetto stesso si chiama `progettoembedded`)
- **@SuppressWarnings("MissingPermission"):** I permessi sono stati controllati correttamente anche se Lint segnala il contrario. Ad ogni modo, è possibile arrivare in tali parti del codice solo dopo che i permessi sono stati forniti, dunque non rappresenterebbe un problema in ogni caso
- **@SuppressWarnings("StaticFieldLeak"):** Non è un leak di nessun tipo. Si è deciso di mantenere un'unica istanza di `readerService` condivisa tra `Activity` e `Fragment` e permettere solo all'`activity` di fare `Bound/Unbound` con essa.