

Breve Descrizione dell'Applicazione

Utilizzo dell'Applicazione

L'applicazione, al suo avvio, si compone principalmente di due EditText (all'interno delle quali verranno stampati i valori, rispettivamente, della latitudine e della longitudine corrente dell'utente) e di un bottone che, alla pressione, si occuperà di inoltrare la richiesta sull'aggiornamento della posizione. Dal momento che l'applicazione utilizza il GPS, come prima cosa devono essere chiesti i permessi a runtime (oltre a quelli inseriti all'interno del Manifest) per utilizzarlo. Di conseguenza, la prima volta che verrà premuto il bottone verrà chiesto se consentire o no questi permessi. In caso di risposta positiva da parte dell'utente, l'applicazione potrà essere utilizzata correttamente per ottenere latitudine e longitudine corrente. Da questo momento in poi, infatti, andando a premere il bottone "Ottieni Posizione" verrà avviata la richiesta sull'aggiornamento della posizione e compariranno una nuova TextView e una ProgressBar per dare all'utente un feedback sul fatto che l'applicazione sia in attesa di ricevere la posizione corrente, e verrà mostrato un ulteriore bottone, "Annulla Richiesta", allo scopo di annullare appunto la richiesta qualora questo sia il desiderio dell'utente. Una volta che la richiesta è andata a buon fine, nelle due EditText verranno impostati i valori di latitudine e longitudine ottenuti e l'utente eventualmente potrà avviare una nuova richiesta per l'ottenimento di una nuova posizione.

Oltre alla versione in Italiano dell'applicazione, è stata prevista anche una versione in lingua Inglese, disponibile nel momento in cui si imposti quest'ultima come lingua del dispositivo.

Inoltre, di seguito è fornita una spiegazione dei principali metodi utilizzati per lo sviluppo dell'applicazione.

Principali Metodi Utilizzati

`getLastLocation()` → All'interno di questo metodo effettuiamo la richiesta per la latitudine e la longitudine corrente e settiamo le due EditText in modo tale che contengano i valori ottenuti dalla richiesta. Poiché la richiesta potrebbe richiedere un po' di tempo, abbiamo inserito un Toast che notifica quando la richiesta è andata a buon fine. Alla fine, chiamiamo **`locationManager.removeUpdates(locationListenerGPS)`** per rimuovere tutti gli update dal Location Listener. Questo metodo viene chiamato nella **`onLocationChanged()`**.

`isLocationEnabled()` → Abbiamo implementato questo metodo in modo tale che verifichi se il GPS è attualmente attivo sul device, tramite il metodo **`isProviderEnabled()`** di LocationManager. Se il GPS non è attivo, questo metodo fa apparire un AlertDialog che lo notifica e che permette all'utente di essere indirizzato alle impostazioni del device nel caso in cui decida di attivarlo in quel momento per poter usufruire delle funzionalità dell'applicazione. Se il GPS invece è attivo, viene mostrato un Toast per notificarlo all'utente, in modo tale che esso sia ulteriormente (dal momento che ha già concesso i permessi) consapevole del fatto che l'applicazione sta utilizzando il GPS. La chiamata di questo metodo viene effettuata sia in **`onProviderEnabled()`**, sia in **`onProviderDisabled()`**.

getLocation() → Questo metodo viene chiamato alla pressione del bottone “Ottieni Posizione”.

Al suo interno innanzitutto verifichiamo se abbiamo i permessi di accesso per il GPS attraverso il metodo ***checkSelfPermission()***. Se non abbiamo i permessi e non sono mai stati negati, richiediamo i permessi per la prima volta tramite ***requestPermissions()***. Se invece non abbiamo i permessi e precedentemente erano stati già negati (e lo verifichiamo tramite ***shouldShowRequestPermissionRationale()***), viene mostrato un AlertDialog per informare l’utente del fatto che aveva già negato i permessi, ma che per continuare necessita di questi ultimi. Dunque gli viene fornita la possibilità di farsi richiedere nuovamente i permessi nel caso in cui preme il bottone positivo dell’Alert. D’altra parte, se i permessi sono già stati concessi, questo metodo verifica se il Provider del GPS è abilitato tramite la chiamata a ***isLocationEnabled()*** e in caso affermativo inoltra la richiesta per la posizione attraverso il metodo ***locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, locationListenerGPS)*** e, una volta ricevuto l’aggiornamento sulla posizione, verrà chiamato il metodo ***onLocationChanged()***, che al suo interno chiamerà a sua volta ***getLastLocation()***.

cancelRequest() → Questo metodo viene chiamato alla pressione del bottone “Annulla Richiesta” (che compare solo nel caso in cui la richiesta sia andata a buon fine e sia attualmente in corso) per annullare la richiesta di aggiornamento sulla posizione . Di conseguenza, al suo interno viene chiamato ***locationManager.removeUpdates(locationListenerGPS)***. Poiché molto spesso la richiesta di aggiornamento sulla posizione impiega molto tempo prima di andare a buon fine, abbiamo ritenuto una buona idea quella di inserire la chiamata a questo metodo anche nella ***onPause()***. Questo per salvaguardare l’utilizzo della batteria nel caso in cui l’utente lasci l’applicazione in background (dimenticandosi eventualmente di aver lanciato una richiesta) e il GPS non riesca a fornire la posizione corrente per molto tempo .