

Scopo progetto

Lo scopo del progetto è quello di sviluppare un'applicazione mobile per il sistema operativo Android che fornisca informazioni meteorologiche per le diverse località. L'applicazione deve consentire agli utenti di visualizzare le informazioni meteorologiche correnti per le località di loro interesse. Inoltre, l'applicazione deve permettere agli utenti di aggiungere nuove località manualmente, utilizzando il GPS per rilevare la posizione corrente dell'utente o cercando la località attraverso la ricerca di testo. L'obiettivo finale dell'applicazione è quello di fornire agli utenti informazioni accurate sulle condizioni meteorologiche attuali.

Requisiti

- Applicazione di tipo *List - Detail*
- Possibilità di aggiungere nuove location manualmente (con popup, nuova schermata, ...)
- Utilizzo del **GPS** per leggere la posizione corrente e mostrarla in lista
- Salvataggio delle location inserite dall'utente su **database** (SQLite)
- Controllo periodico (tramite **Background Service**) delle temperature; invio di **notifiche** se la temperatura locale scende / sale sopra una certa soglia

Progetto

Il progetto è stato scritto in Java e abbiamo utilizzato Android Studio come ambiente di sviluppo integrato (IDE).

Tecnologie utilizzate

1. **SQLite**: SQLite è un database relazionale leggero e veloce. SQLite è una scelta comune per le app mobili Android in quanto è una tecnologia di database locale che non richiede un server separato e non richiede una connessione a Internet per funzionare.
2. **SmartLocation**: SmartLocation è una libreria Android open source che offre una soluzione semplice e veloce per accedere alle informazioni sulla posizione come latitudine, longitudine, velocità, altitudine, ecc. Offre funzionalità avanzate come il geocoding inverso (ovvero la conversione di coordinate GPS in un indirizzo fisico).

Architettura Progetto

Il progetto è stato suddiviso in tre principali cartelle che sono:

- La cartella "**manifest**" contiene un solo file, ovvero il file AndroidManifest.xml. Questo file è obbligatorio in ogni progetto Android e contiene le informazioni dell'applicazione, come il nome dell'applicazione, le icone, le autorizzazioni richieste, l'attività principale, i servizi, i ricevitori di messaggi, le librerie esterne, ecc. Inoltre, il file manifest definisce le caratteristiche minime del dispositivo richieste dall'applicazione, ad esempio la versione di Android.
- La cartella "**java**" contiene tutti i file sorgente scritti in linguaggio Java. Questi file rappresentano la logica dell'applicazione e includono classi, interfacce, metodi, ecc.
- La cartella "**resource**" dove contiene tutte le risorse non codice dell'applicazione.

Nello specifico la cartella java è stata suddivisa in sottocartelle.

- cartella "**activities**" contiene le due attività dell'applicazione. MainActivity.java è la pagina principale dell'applicazione, in cui viene visualizzato un elenco di città. In questa attività, si ha una RecyclerView che elenca le città disponibili. Quando l'utente seleziona una città dall'elenco, viene avviata la DetailActivity per mostrare i dettagli del meteo attuale della città selezionata.
- La cartella "**model**" contiene le classi che definiscono i dati dell'applicazione (LocationModel, WeatherModel).
- La cartella "**database**" contiene tutte le classi e le interfacce necessarie per gestire la persistenza dei dati dell'applicazione in Locale utilizzando SQLite.
- La cartella "**fragment**" contiene tutti i frammenti dell'applicazione.
- La cartella "**worker**" contiene tutte le classi che gestiscono le attività in background dell'applicazione (handler notifiche).

Interfaccia utente

Nell'immagine 1.0 possiamo vedere la schermata principale dell'applicazione dove è possibile visualizzare la lista delle città salvate dall'utente.

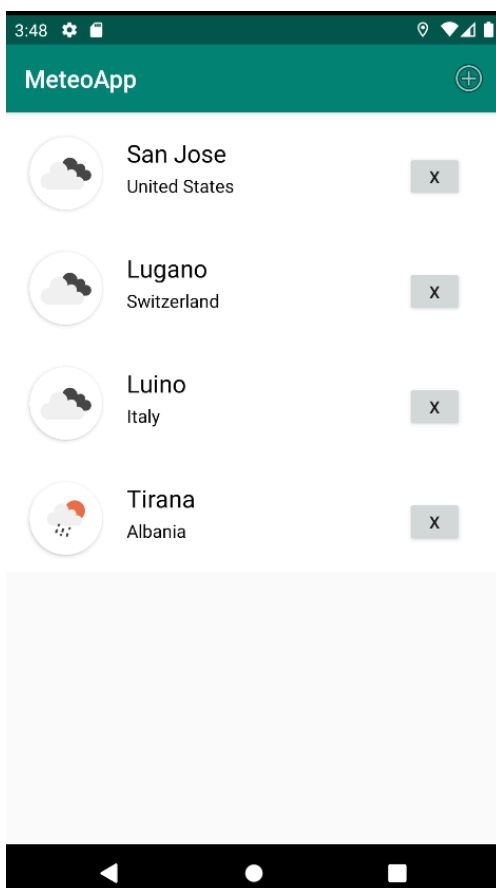
Nella prima posizione abbiamo la posizione attuale, mentre dalla seconda posizione in poi della lista abbiamo le città cercate e salvate dall'utente.

E' possibile aggiungere una nuova città cliccando nell'icona + in alto a destra. Dove si aprirà un pop up di aggiunta (vedi immagine 2.0)

Inoltre è possibile rimuovere le città cliccando semplicemente sulla x a destra della città la si vuole rimuovere.

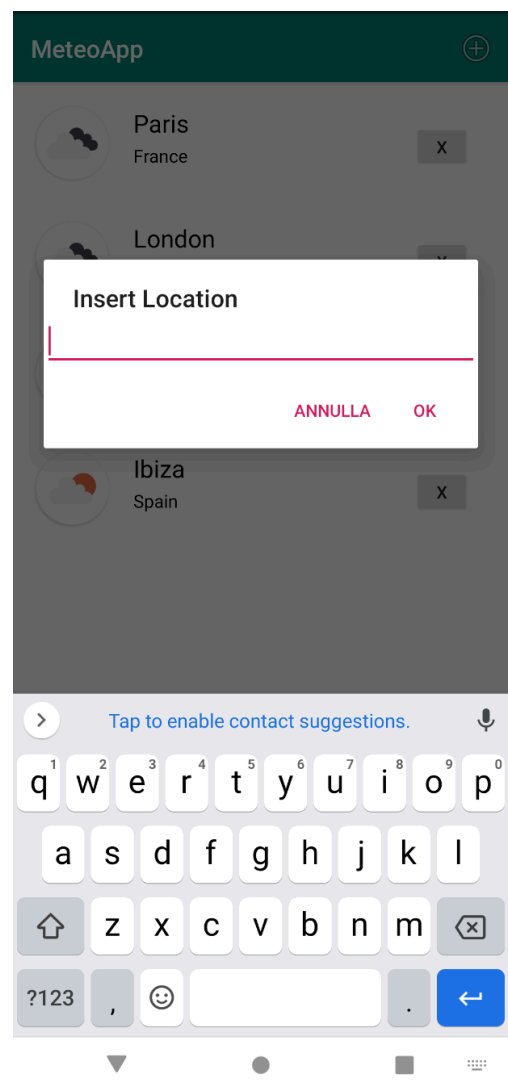
Infine al click del nome della città si aprirà una nuova schermata di dettagli meteo (vedi immagine 3.0)

Schermata Home



1.0

Schermata inserimento location



2.0

Schermata details

