



## Cronometro con conteggio alla rovescia

Utilizzando l'approccio visto a lezione, si realizzi un cronometro che effettui un conteggio alla rovescia e, giunto al termine, emetta un suono.

L'applicazione sarà basata su un singola schermata, all'interno della quale troveranno posto tutti gli elementi necessari al suo funzionamento: l'indicazione del tempo residuo, i pulsanti necessari al suo avvio/arresto, quanto necessario per impostare il tempo iniziale.

Nella sua apparente semplicità, questa applicazione nasconde alcune sfide:

- Sul piano dell'**interazione**, come posso impostare in modo agevole e facilmente intuibile il tempo iniziale? Devo prevedere o meno la possibilità di una pausa nel conteggio? Quanto a lungo deve suonare, quanto il conteggio arriva a zero? L'indicazione del tempo residuo è digitale o analogica?
- Sul piano della **grafica**, come impostare la schermata perché sia al tempo stesso gradevole, leggibile, di facile e comoda interazione (posizione e distanza reciproca degli elementi interattivi) sia per persone destre che mancine? I pulsanti hanno un'etichetta testuale o un'icona?
- Sul piano della **programmazione**, come posso aggiornare la schermata allo scorrere del tempo? Come suddivido i compiti tra le diverse classi presenti nel programma? Come faccio a riprodurre dei suoni?
- Sul piano della **messa in campo**, come posso installare l'app realizzata su un telefono reale? Come la cancello?
- Sul piano della **gestione del progetto**, riusciamo ad organizzare il lavoro in modo da concluderlo entro la prossima settimana, coinvolgendo al tempo stesso tutti i membri del gruppo? Come possiamo documentare quanto svolto, così da permettere a tutti di capire perché, cosa e come abbiano fatto gli altri membri del gruppo, così da rendere il sapere patrimonio comune (visto che l'esame sarà uguale per tutti)?

**Attenzione:** tra un prodotto completo in tutte le sue funzionalità e il nulla esistono numerosi passi intermedi. Procedete per affinamenti successivi, scegliendo di volta in volta il cosiddetto Minimum Viable Product, ovvero il sottoinsieme più piccolo possibile delle funzionalità che riuscite a realizzare con caratteristiche appena sufficienti per essere utilizzabile dai primi clienti, così da poterne raccogliere il feedback. Ad ogni successiva iterazione, rivedete quanto fatto alla luce delle informazioni raccolte ed aggiungete qualche nuova caratteristica, che nel frattempo avete studiato e preparato.

## Note per l'implementazione

L'applicazione conterrà due classi: una, chiamata **CountdownActivity** (che estende `AppCompatActivity`), conterrà la logica di presentazione della schermata ed una seconda, chiamata **CountdownViewModel** (che estende `ViewModel`), destinata a contenere gli oggetti osservabili (`MutableLiveData<T>`) che costituiscono il suo stato (tempo residuo, conteggio attivo/fermo, ...) e ad offrire i metodi necessari a modificarlo (`start()`, `stop()`, ...).

Per poter correttamente utilizzare il collegamento tra `viewModel` e `activity`, occorre includere, all'elenco delle librerie del progetto, quella denominata "androidx.activity:activity-ktx:1.7.0", da inserire nel file *build.gradle.kts* (*Module :app*).