*Digital Text Suite*

Analisi di progetto

# Linguaggio usato

Il linguaggio utilizzato per lo sviluppo del progetto è Kotlin. È stato scelto tale linguaggio sia perché consigliato da Google per lo sviluppo di applicazioni Android, sia perché è un linguaggio molto intuitivo, che viene incontro alle esigenze del programmatore attraverso costrutti semplici da utilizzare.

# Scelte architetturali

Nello sviluppo dell’applicativo, si è cercato di applicare il pattern architetturale MVVM (Model – View – ViewModel), in quanto è stato ritenuto più adatto per lo sviluppo di applicazioni in ambiente Android, rispetto ad altri pattern architetturali come ad esempio MVC. Infatti, si presta bene alla gestione di continui combiamenti in oggetti di View, ad esempio a seguito di accessi ad un database; inoltre, è molto efficiente ed efficace per la comunicazione tra Fragment. Dunque, la scelta è stata quasi automatica, avendo progettato l’applicazione in maniera fragment-oriented.

Inoltre, si è cercato di perseguire i paradigmi della buona progettazione e programmazione Object Oriented, tentando, ove possibile, un approccio orientato all’uso di interfacce e che rendesse maggiormente possibile il riuso del codice.

Ad esempio, è stata riutilizzata la classe “CameraFragment”, precedentemente sviluppata nell’ambito del progetto “EffectsCam”, rendendola astratta, così da poterne usufruire in diversi punti dell’applicazione, sfruttando il meccanismo dell’ereditarietà.

Un altro esempio è costituito dall’ampio utilizzo di Dialog, riusati in diverse occasioni anche per scopi differenti. Ciò è stato reso possibile grazie all’utilizzo di listeners, che rendono i Dialog il più personalizzabili possibile.

Anche l’utilizzo di interfacce contribuisce a rendere il codice dell’applicazione più riusabile e a mantenere, per quanto possibile, un livello di coupling tra le classi non eccessivamente elevato, seppur ciò risulti particolarmente complicato da ottenere in ambiente Android, essendoci, per la natura stessa delle applicazioni mobile, un’elevata dipendenza dai componenti di View.