



# MEMOME

Relazione relativa al progetto per il corso Mobile Programming

Anno Accademico 2018-2019

Sviluppato da: Andrea Efficace, Ionut Marian Motoi, Lorenzo Rossi

Presentato il: 16 Settembre 2019

# Riepilogo Esecutivo

## **Introduzione**

Una nota è un breve testo esplicativo ormai fondamentale nella vita di tutti i giorni per annotare pensieri, eventi o più in generale una qualsiasi informazione da non dimenticare. Con la pubblicazione di applicazioni di note per i dispositivi mobili, le note permettono di ottenere una maggiore personalizzazione rispetto alla versione cartacea.

## **SCELTA DEL PROGETTO**

Tra le 3 opzioni proposte, siamo stati contenti che la scelta del docente sia ricaduta sul progetto Memo me.

Una volta letta la traccia, infatti, abbiamo subito pensato di poter valorizzare il progetto sia in termini di grafica che in termini di implementazione della gestione delle note, mantenendo una certa attenzione sulla facilità d'utilizzo della nota.

## **FASI DI PROGETTAZIONE**

Nelle seguenti pagine verranno descritte le varie fasi della progettazione, ovvero i problemi sorti prima e dopo l'implementazione del codice effettivo con le relative soluzioni e analisi dei requisiti fino ad arrivare ai casi di utilizzo.

# PROGETTAZIONE

In questa sezione viene illustrata la fase di progettazione dell'applicazione Android MemoMe.

## Analisi dei requisiti

- L'applicazione deve essere supportata dalla maggioranza de dispositivi Android (API >23);
- L'applicazione deve poter essere fruita in lingue diverse;
- L'applicazione deve avere un titolo max di 32 caratteri e contenere un testo di max 255 caratteri;
- L'applicazione deve tenere traccia della data di creazione e di ultima modifica;
- L'applicazione MemoMe deve poter scrivere e modificare le note;
- L'applicazione deve poter cambiare il colore delle note che deve corrispondere al colore dello sfondo nella lista delle note;
- L'applicazione deve poter consentire l'aggiunta di una icona ad ogni nota;
- L'applicazione deve presentare le note attraverso una lista ordinabile per colore;
- L'applicazione deve permettere la modifica del titolo, del contenuto, del colore e dell'icona;
- Facilità di utilizzo;

## Problemi riscontrati

I problemi riscontrati nella progettazione di questa applicazione sono i seguenti:

- 1-** Corretta gestione dell'interfaccia grafica dell'applicazione e delle librerie *androidx* e *MaterialDesign*;
- 2-** Aggiunta della selezione dei colori e delle icone;
- 3-** Corretta gestione dei *Fragment*;
- 4-** Reperibilità di librerie funzionanti e adatte ai nostri scopi;
- 5-** Ordinamento delle note;
- 6-** Sistema di salvataggio delle note

## **Soluzioni adottate:**

### **1-Corretta gestione dell'interfaccia e librerie androidx e MaterialDesign**

Uno dei problemi principali è stato quello di come organizzare l'interfaccia utente per ottenere un *Material Design*, mantenendo un facile utilizzo della applicazione. Per fare ciò è stato svolto un approfondito studio da documentazione dal sito *Android Developers* e tramite approfondite ricerche sull'utilizzo dei vari elementi grafici sul sito *Stack Overflow*. In particolare sono state utilizzate *Card View* per permettere la visualizzazione delle note su una *Recycle View*, *Material Button*, *Material TextInputLayout* per implementare una *textInputEdit* che consente di ottenere il totale controllo dell'ambiente grafico della casella di testo.

### **2- Aggiunta della selezione dei colori e delle icone**

Data la struttura dell'applicazione a cui siamo giunti al momento dell'implementazione di queste due funzionalità, siamo stati costretti a ristrutturare il database per aggiungere dei campi (*Icone* e *colori*) che permettessero di memorizzare lo stato corrente dell'oggetto, secondo le specifiche.

### **3- Corretta gestione dei Fragment**

L'uso dei *Fragment* è stato destinato all'implementazione delle funzionalità di selezione icona e colore.

La prima è stata implementata completamente dal gruppo, scegliendo il posizionamento più accurato in base al Layout raggiunto e implementando correttamente il suo ciclo di vita per evitare artefatti grafici dopo il suo utilizzo.

### **4- Reperibilità di librerie utili alla nostra applicazione**

Tramite il sito *Android Arsenal* abbiamo implementato la libreria *colorpicker* che consente di aprire una schermata per la selezione dei colori. In aggiunta consente di impostare il colore selezionato dalla schermata già implementata nella libreria.

### **5-Ordinamento delle note**

Abbiamo impiegato diversi metodi per soddisfare le specifiche riguardo all'ordinamento delle note tramite *Collections.sort*. Questa classe permette di ordinare una lista di oggetti confrontando i loro attributi (*testo, data, colore*).

### **6-Sistema di salvataggio delle note**

Il sistema di salvataggio note è reso possibile dall'implementazione della libreria *Room Persistence* che pone un ulteriore livello di astrazione sopra il sistema *SQLite* per permettere accessi più robusti al database mantenendo allo stesso tempo tutte le sue potenzialità. Tale implementazione è stata preferita data la facilità di manipolazione dei dati.

## SCENARI DI INTERAZIONE

In questa sezione l'attenzione viene focalizzata su alcuni scenari di interazione relativi all'applicazione MemoMe.

### **Avvio dell'applicazione**

L'utente al primo avvio può aggiungere una nuova nota tramite l'apposito pulsante. Nei successivi accessi all'applicazione, verrà visualizzata la lista delle note precedentemente inserite con la possibilità di ordinarle e ricercarle con le specifiche richieste.

### **Inserimento di una nota**

L'utente può modificare il titolo e il contenuto della nota cliccando sulle apposite caselle di testo. Inoltre può modificare con gli appositi bottoni il colore dello sfondo della nota e aggiungere eventualmente una icona personalizzata.

### **Cancellazione di una nota**

All'utente è data la possibilità di cancellare le note da lui inserite nelle seguenti 2 modalità:

- 1- Dalla *Main Activity* tramite lo scorrimento verso sinistra. Inoltre l'utente dispone di una funzionalità di ripristino della nota eliminata in caso di errore, implementata tramite una *SnackBar* con tasto;
- 2- Dalla *EditActivity* tramite un bottone che implementa la cancellazione tramite un *DialogAlert*;

## **Versione in Italiano e Inglese**

L'applicazione MemoMe è disponibile nella versione italiana o inglese a seconda della lingua predefinita del sistema operativo.