

# Programmazione Dispositivi Mobili

Proposta di progetto

---

## SIMPLESAMPLE

## GRUPPO

Informazioni sul gruppo di laboratorio.

<b>Nome del gruppo:</b>	SimpleSample Dev Team
<b>Componenti:</b>	• Giorgio Gamba, matricola 833313

## DATE

Le date principali del documento.

<b>Data di sottomissione della proposta di progetto</b>	25 maggio 2021
<b>Data di accettazione della proposta di progetto</b>	29 Maggio 2021

## DESCRIZIONE BREVE

Descrizione della App in una frase.

Muovi i primi passi nella composizione musicale creando tracce uniche realizzate con i suoni dal mondo che ti circonda!

## DEFINIZIONI

Di seguito la definizione dei termini, abbreviazioni e acronimi utilizzati.

Termine	Definizione
<b>Sample</b>	Breve suono registrato tramite dispositivo di cattura o estratto da una traccia audio. Viene chiamato anche "campione"
<b>Traccia</b>	File audio musicale rappresentante un brano
<b>Campionatore</b>	Strumento utilizzato a partire dagli anni '80 in grado di manipolare samples e riprodurli a discrezione dell'utente. Esempi di campionatori sono i prodotti della famiglia Akai MPC

<b>Beat</b>	Composizione musicale contenente suoni percussivi, armonici e/o melodici
<b>Sequencer</b>	Strumento per la creazione e riproduzione di segnali per il controllo di vari strumenti musicali. Tipicamente tale strumento veniva usato per comandare anche i campionatori al fine della riproduzione automatica dei samples contenuti al loro interno
<b>UI</b>	Sigla per indicare "User Interface", ovvero l'interfaccia utente
<b>MVC</b>	Pattern di progettazione "Model-View-Controller" che disaccoppia i dati dell'applicazione dalla vista utente tramite l'intermediazione di una struttura "controller"
<b>Widget</b>	Componente base per l'implementazione di applicazioni tramite Flutter

# SOMMARIO

Gruppo.....	2
Date .....	2
Descrizione breve .....	2
Definizioni .....	2
Sommario .....	4
1. Contesto del progetto.....	5
1.1. Situazione attuale .....	5
1.2. Benefici e creazione di valore .....	5
1.3. Obiettivi del progetto.....	5
2. Profilo del progetto .....	6
2.1. Ambito del progetto .....	6
2.2. Profilo della soluzione da realizzare .....	6
3. Vincoli e assunti.....	8
3.1. Vincoli temporali.....	8
3.2. Vincoli tecnologici.....	8
3.2.1. Privilegi .....	8
3.3. Eventuali assunti.....	8

# 1. CONTESTO DEL PROGETTO

---

## 1.1. Situazione attuale

Attualmente, sul Play Store come sull'App Store esistono diverse soluzioni per il campionamento e riproduzione di samples, ma non esistono praticamente soluzioni cross-platform se non a pagamento. Inoltre, molte delle app disponibili risultano essere difficili da utilizzare per una persona che per la prima volta si affaccia al mondo del campionamento o della composizione musicale in generale, essendo che forniscono features già avanzate a volte non semplici da sfruttare per i neofiti e i non musicisti. Inoltre, queste applicazioni non forniscono sistemi di condivisione o salvataggio su dispositivi esterni per permettere di riutilizzare tali dati in altri ambiti.

## 1.2. Benefici e creazione di valore

SimpleSample si propone di permettere a tutti i neofiti di potersi interfacciare in maniera semplice alla composizione musicale, allo stesso modo in cui i campionatori MPC hanno permesso di creare nuovi panorami musicali molti anni fa, quali HipHop e Elettronica.

Inoltre, utenti esperti possono trovare in SimpleSample un motivo di svago o uno strumento utile a raccogliere idee da applicare in altri ambienti tramite il supporto ai servizi cloud. Esempio di utilizzo potrebbe essere quello di raccogliere suoni all'esterno mentre si sta passeggiando in città per poi riutilizzarli nel proprio home-studio durante una vera sessione di composizione e produzione musicale. Tutti gli utenti dell'applicazione potranno condividere le proprie registrazioni e ascoltare quelle di altri, con la possibilità, quindi, di ascoltare e riutilizzare a proprio piacimento i samples prodotti da altri utenti.

## 1.3. Obiettivi del progetto

Elenco di aspetti specifici e verificabili che realizzerete nella vostra App e che la renderanno "appetibile".

<b>Aspetto 1</b>	Sistema di registrazione immediato e di riproduzione dei suoni stessi
<b>Aspetto 2</b>	Definizione della UI in Flutter e conseguente support cross-platform
<b>Aspetto 3</b>	Possibilità di utilizzo sia di dati catturati con microfono che prelevati dal filesystem del telefono
<b>Aspetto 4</b>	Salvataggio dei dati raccolti nel filesystem del dispositivo
<b>Aspetto 5</b>	Uso di differenti layout per le varie funzionalità dell'applicazione
<b>Aspetto 6</b>	Servizio di collegamento a strumenti di cloud per il salvataggio dei dati
<b>Aspetto 7</b>	Sistema di iscrizione alla piattaforma con username univoco per ogni utente
<b>Aspetto 8</b>	Possibilità di condivisione dei propri samples (o delle composizioni create tramite il sequencer) con altri utenti della piattaforma
<b>Aspetto 9</b>	Sistema di notifica all'autore di un campione qualora un altro utente iscritto a SimpleSample decidesse di utilizzare un suo campione
<b>Aspetto 10</b>	Possibilità di mantenere traccia dei file prodotti da altri utenti tramite sistema di "preferiti"

## 2. PROFILO DEL PROGETTO

---

### 2.1. Ambito del progetto

SimpleSample si propone di astrarre e simulare parte del comportamento di un vecchio campionatore Akai MPC, strumento utilizzato molto a partire dagli anni '80 che ha permesso la nascita di vari generi musicali. Il processo di utilizzo di tale strumento parte dal taglio di estratti da tracce audio, ottenendo i cosiddetti "samples", seguita dall'assegnamento di tali campioni a una tastiera 4x4 che ne permette la riproduzione in tempo reale a discrezione dell'utente, creando di fatto una nuova traccia audio.

La mia applicazione si propone di riprodurre lo stesso concetto, permettendo però all'utente di poter registrare suoni raccolti dall'ambiente che lo circonda, in modo da poterli sia suonare con l'app stessa sia per creare una sorta di archivio di samples da poter riutilizzare anche in altri ambienti, come i sempre più presenti studi casalinghi degli appassionati di produzione musicale. Questo passaggio ad un altro ambiente di lavoro sarà favorito dall'integrazione con servizi di cloud quale Dropbox, molto usato in ambito musicale. Questo software si propone quindi anche come strumento di supporto per i produttori musicali sotto forma di raccoglitore di suoni.



Inoltre, l'applicazione implementerà un sequencer per il controllo dei campioni in uso, al fine di creare dei semplici beat che permettano all'utente di testare l'usabilità e la qualità dei samples precedentemente registrati.

Al fine di permettere una facile interazione soprattutto per i neofiti, la UI verrà mantenuta il più semplice possibile.

Verrà inoltre sviluppata un sistema di iscrizione alla piattaforma SimpleSample in cui ogni utente potrà creare un proprio account con nome utente univoco e decidere se condividere i propri samples (o le composizioni create tramite il sequencer) con gli altri membri della piattaforma. Sarà quindi implementato un sistema di navigazione dei file audio globalmente disponibili con possibilità di salvare quelli che si vuole riascoltare tramite un sistema di "preferiti".

Applicazioni di questo tipo offrono anche la possibilità di implementare molte funzionalità aggiuntive, come un mixer per il controllo dei volumi dei campioni, un serie di effetti audio.

### 2.2. Profilo della soluzione da realizzare

SimpleSample sarà dotata di due schermate principali:

- "Sampler": conterrà una griglia di tasti che permetteranno di registrare suoni (eventualmente prelevabili dal filesystem) e riprodurli in modo interattivo
- "Sequencer": conterrà un'interfaccia simile a quella di un sequencer reale che permetterà di riprodurre in maniera automatizzata i suoni raccolti tramite la schermata "Sampler"

Verrà, inoltre, implementato un menù dedicato all'utente nel quale potranno venire gestite altre funzionalità, quali eventuali impostazioni di utilizzo dell'applicazione, informazioni di accesso al sistema di cloud, file audio preferiti o resi disponibili agli altri utenti.

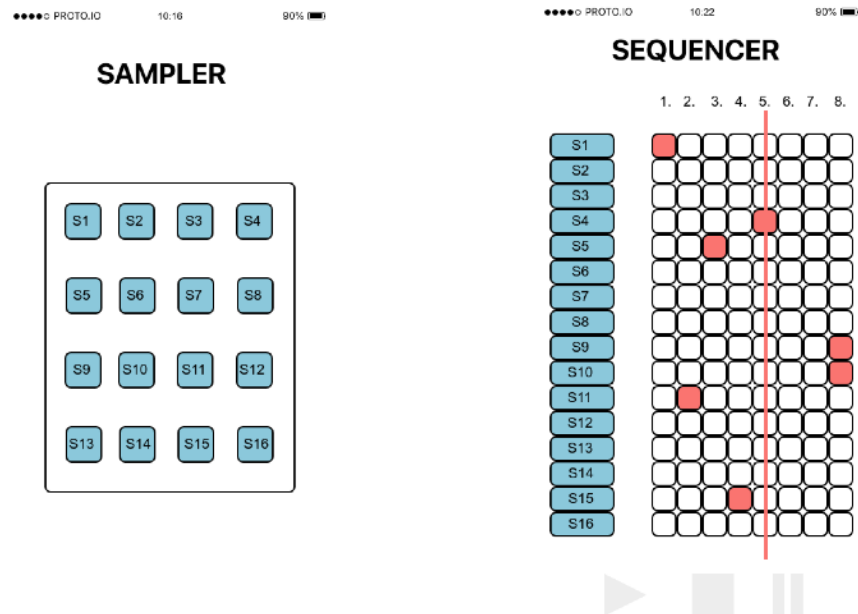
Al fine di poter esplorare le registrazioni condivise dagli altri utenti, verrà realizzato un sistema di ricerca globale tra tutti i dati resi disponibili sulla piattaforma.

Il software si baserà sul pattern di progettazione MVC, il quale consentirà di poter separare l'interfaccia utente dal motore audio e dai campioni registrati. Questo permetterà di dividere il lavoro dell'interfaccia visiva dal Modello, che ha il ruolo di gestire i dati audio.

L'applicazione verrà sviluppata secondo un approccio "Test Driven Development" (TDD) facendo utilizzo di sistemi di testing secondo i seguenti livelli con i relativi strumenti usati:

- Unit Test: fa uso del pacchetto Flutter "test", framework ufficiale di Flutter
- Widget Test: utilizza il pacchetto "flutter\_test", framework basato a sua volta sul pacchetto "test" di Flutter
- Integration Test: utilizza il pacchetto Flutter "integration\_test" (versione Flutter del framework di testing Android "Espresso")

Qui di seguito sono presentate due possibili implementazioni delle interfacce utente delle due schermate principali, "Sampler" e "Sequencer"



## 3. VINCOLI E ASSUNTI

### 3.1. Vincoli temporali

Le principali date previste per lo sviluppo dell'applicazione sono:

<b>1 Giugno 2021</b>	Inizio dello sviluppo dell'applicazione
<b>20 Giugno 2021</b>	Rilascio della prima release funzionante che implementa almeno una parte delle funzionalità elencate
<b>5 Luglio 2021</b>	Sessione d'esame in cui avverrà la discussione del laboratorio

### 3.2. Vincoli tecnologici

Il progetto verrà realizzato in Flutter al fine di poter fornire un utilizzo cross-platform, punto cardine dei motivi della realizzazione del progetto. Per il mantenimento dei dati disponibili a tutti gli utenti si farà utilizzo della piattaforma di storage di Firebase. Come anche specificato nella versione di Flutter 2.2 utilizzata, per Android si svilupperà dalla versione minima KitKat (API 19), mentre per iOS verranno supportate tutte le versioni dalla 9 (compresa) in avanti.

#### 3.2.1. Privilegi

SimpleSample richiederà le seguenti autorizzazioni:

<b>Login a sistema cloud (facoltativo)</b>	Questo privilegio permetterebbe all'utente di poter raccogliere all'interno di un sistema cloud tutti i file registrati durante l'utilizzo dell'applicazione, portando diversi vantaggi: <ul style="list-style-type: none"><li>• Risparmio di spazio all'interno del dispositivo</li><li>• Facile reperimento da un altro dispositivo (come il computer di uno studio di produzione)</li></ul>
<b>Manifest.permission.RECORD_AUDIO (Android) / NSMicrophoneUsageDescription (iOS)</b>	Permette al software di poter raccogliere informazioni audio tramite il microfono del dispositivo
<b>android.permission.WRITE_EXTERNAL_STORAGE (Android) / Relativa controparte iOS</b>	Questo privilegio consente all'applicazione di poter sia salvare le registrazioni all'interno del dispositivo (o sull'eventuale supporto di memoria esterno) che di poter reperire i file audio da esso
<b>android.permission.INTERNET / Relativa controparte iOS</b>	Privilegio per accesso alla rete e a Firebase

### 3.3. Eventuali assunti

- I servizi di Cloud implementati rimangono disponibili
- Il servizio di storage di Firebase rimanga disponibile



