



Todo *Todo*, Todo, © gennaio 2020.

Per le immagini riportate in questa relazione :

Orchestra: si veda la figura ?? a pagina ??

Fonte: [https://upload.wikimedia.org/wikipedia/commons/3/39/MIT0\\_Orchestra\\_Sinfonica\\_RAI.jpg](https://upload.wikimedia.org/wikipedia/commons/3/39/MIT0_Orchestra_Sinfonica_RAI.jpg)

Foto originale di MITO SettembreMusica, 2008

© [https://upload.wikimedia.org/wikipedia/commons/3/39/MIT0\\_Orchestra\\_Sinfonica\\_RAI.jpg](https://upload.wikimedia.org/wikipedia/commons/3/39/MIT0_Orchestra_Sinfonica_RAI.jpg)

Licenza: Creative Commons

Tutti i marchi riportati appartengono ai legittimi proprietari.





# RINGRAZIAMENTI

*Todo*

*Padova, gennaio 2020*

Matteo Marchiori, Giovanni Peron



# INDICE

Frontespizio	i
Ringraziamenti	iv
Indice	vii
Elenco delle figure	ix
Elenco delle tabelle	xi
1 INTRODUZIONE	1
1.1 Todo	1
1.1.1 Todo	1
1.2 Utenti target	1
2 DESIGN DELL'APPLICAZIONE	3
2.1 Usabilità	3
2.2 Caratteristiche salienti	3
2.3 Android	3
2.4 i-OS	3
2.5 Le gesture	3
3 GAMIFICATION	5
4 IMPLEMENTAZIONE	7
4.1 Ambiente di sviluppo	7
4.2 Il framework Flutter	7
4.2.1 Decrizione	7
4.2.2 Vantaggi e svantaggi	8
4.2.3 Comparazione con altri framework	8
4.3 Strumenti di test usati	8
4.4 Performance	8
5 TODO	9
Bibliografia	11





## ELENCO DELLE FIGURE

Figura 1	Architettura di Flutter	8
----------	-------------------------	---



## ELENCO DELLE TABELLE



# 1 | INTRODUZIONE

*T*odo.

## 1.1 TODO

### 1.1.1 Todo

## 1.2 UTENTI TARGET



# 2 | DESIGN DELL'APPLICAZIONE

2.1 USABILITÀ

2.2 CARATTERISTICHE SALIENTI

2.3 ANDROID

2.4 I-OS

2.5 LE GESTURE





# 3 | GAMIFICATION



# 4

## IMPLEMENTAZIONE

In questa sezione verranno descritte tutte le tecnologie utilizzate per lo sviluppo dell'applicazione.

### 4.1 AMBIENTE DI SVILUPPO

Per lo sviluppo dell'applicazione è stato utilizzato il sistema di controllo di versione Git, la realizzazione del progetto è stata eseguita da i seguenti calcolatori:

- OS: Windows 10 Home 64bit , Ubuntu 16.04, MacOS Catalina
- IDE: Microsoft Visual Studio Code, Android Studio, XCode

### 4.2 IL FRAMEWORK FLUTTER

#### 4.2.1 Decrizione

Flutter è un framework cross-platform cross-compiled realizzato da Google, permette di realizzare applicazioni native eseguibili sia su Android che su iOS realizzando un singolo progetto in linguaggio Dart. La principale caratteristica di Flutter è che la User Interface è composta unicamente da componenti Widget immutabili. I widget possono definire elementi strutturali (barre, pulsanti, menù), stilistici (font, colori) o di layout (padding), e così via; in pratica "tutto è un Widget". Possono essere StatefulWidget, ovvero possedere uno stato ed essere quindi in grado di rispondere alle interazioni dell'utente, oppure StatelessWidget ovvero elementi statici.

Flutter è realizzato in C, C++, Dart e dal motore grafico Skia. Le componenti vengono definite in Dart e re-renderizzate grazie ad un motore implementato in C++ che utilizza le librerie grafiche Skia. Il codice Dart viene compilato in codice nativo utilizzando la compilazione AOT (ahead-of-time) e viene eseguito grazie alla Dart Virtual machine, il motore C/C++ invece viene compilato con NDK Android o con LLVM (Low Level Virtual Machine) in iOS, permettendo la compilazione in codice nativo.

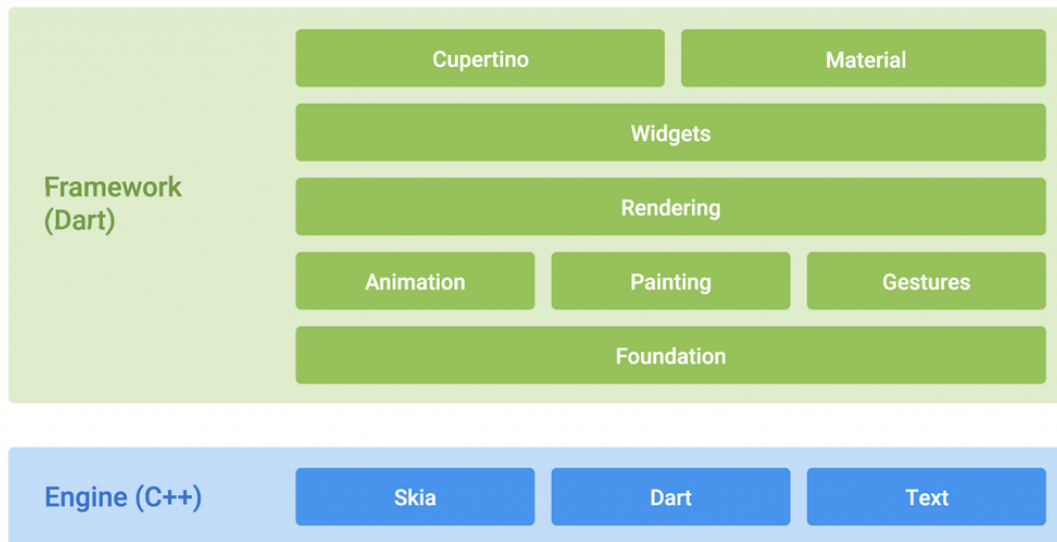


Figura 1: Architettura di Flutter

#### 4.2.2 Vantaggi e svantaggi

I principali vantaggi che il framework Flutter possiede sono:

- **Hot reload:** consente di vedere e testare rapidamente i cambiamenti eseguiti sul codice mentre l'applicazione è in esecuzione, quindi velocizza lo sviluppo;
- **Widgets:** possono essere utilizzati facilmente e sono definiti in stile Material design e Cupertino permettono quindi la realizzazione di un'interfaccia espressiva e flessibile e la loro riusabilità rende lo sviluppo più rapido;
- **Performance native:** il framework essendo cross-compiled garantisce performance al livello di app native;
- **Documentazione:** Flutter è ben documentato e possiede una community di utilizzatori attiva, inoltre è facilmente integrabile con Google Firebase il quale fornisce una rapida implementazione per l'archiviazione dei dati dell'applicazione e delle funzionalità di autenticazione degli utenti;
- **Facilmente integrabile in diversi ambienti di sviluppo:** sono disponibili vari plugins per il supporto in diversi IDE.

#### 4.2.3 Comparazione con altri framework

*Todo.*

### 4.3 STRUMENTI DI TEST USATI

### 4.4 PERFORMANCE

# 5 | TODO

*T*odo.



## BIBLIOGRAFIA