

# Report

Diego Roberto Ammirabile 0001021216  
diego.ammirabile@studio.unibo.it

Febbraio 2025

## Indice

### Panoramica del progetto

#### Obiettivi

**Dettagli di implementazione** Il progetto è stato realizzato attraverso grazie al framework **Ionic** combinato con il framework **Vue**. Quando possibile è stato utilizzato il **Capacitor** per accedere alle funzionalità native dei dispositivi mobili, questo non è sempre stato possibile a causa di limitazioni del framework. Per avere una migliore gestione dei tipi è stato utilizzato **TypeScript**. Il progetto è stato inizializzato con **Vite**.

**Struttura del codice** Il codice è stato organizzato in componenti Vue, in modo da rendere il codice più modulare e manutenibile. Ogni componente è stato progettato per essere il più possibile indipendente dagli altri, in modo da poter essere riutilizzato in altri contesti. Esistono tuttavia componenti che non trovano applicazione in altri contesti se non quello del sovra-componente, tali componenti però gestiscono parti di logica differente da quella del sovra-componente, per cui è stato ritenuto utile mantenerli separati. Il filesystem ha quindi la seguente struttura:

- **android/** contiene i file necessari per la compilazione dell'applicazione per Android
- **dist/** contiene i file compilati
- **node\_modules/** contiene le dipendenze del progetto
- **public/** contiene i file statici
- **src/** contiene il codice sorgente
  - **components/** contiene i componenti Vue
  - **router/** contiene il router
  - **theme/** contiene i file di stile
  - **utils/** contiene le utility

- `views/` contiene le viste
- `App.vue` è il file principale
- `interfaces.ts` contiene le interfacce
- `main.ts` è il file principale
- `variables.json` contiene le variabili, di fatto solo l'URL dell'API
- `vite-env.d.ts` contiene le variabili di ambiente
- `capacitor.config.ts` contiene la configurazione di Capacitor
- `cypress.config.ts` contiene la configurazione di Cypress
- `index.html` è il file principale
- `ionic.config.json` contiene la configurazione di Ionic
- `LICENSE` contiene la licenza
- `package-lock.json` contiene le dipendenze con le versioni esatte
- `package.json` contiene le dipendenze
- `ProjectsLAM2024.pdf` contiene il progettato
- `README.md` contiene le informazioni sul progettato
- `tsconfig.json` contiene la configurazione di TypeScript
- `tsconfig.node.json` contiene la configurazione di TypeScript per Node
- `vite.config.ts` contiene la configurazione di Vite

**Interfaccia utente** La prima schermata che si presenta all'utente è quella di login,


```
export interface User {
  ^^Id: number;
  ^^Iname: string;
  ^^Iemail: string;
  ^^Iphone: string;
}
```

**Conclusioni**

Login

---

Username angolo180

Password ●●●●●●●● 

☐ Remeber me

[LOGIN](#)


Don't have an account? Singup [here](#)

Figura 1: Schermata di login

Sign Up

---

Username

Password 

**SIGN UP**

Already have an account? [Login](#)

Figura 2: Schermata di registrazione

0:00 / 1:00



  
Registra

  
Esplora

  
Cronologia

Figura 3: Schermata di registrazione audio

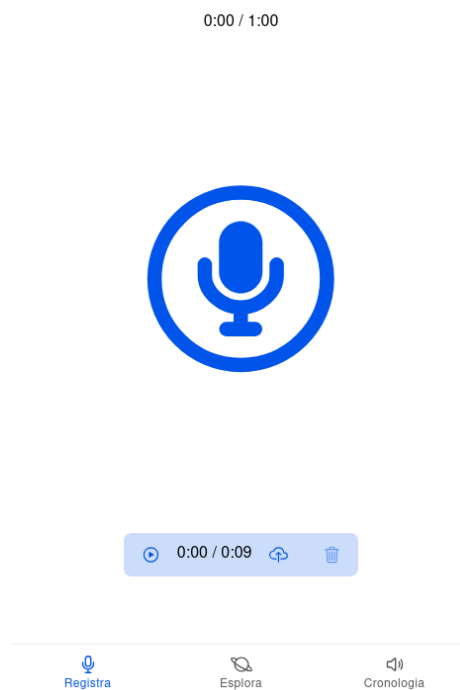


Figura 4: Schermata di registrazione audio con audio registrato

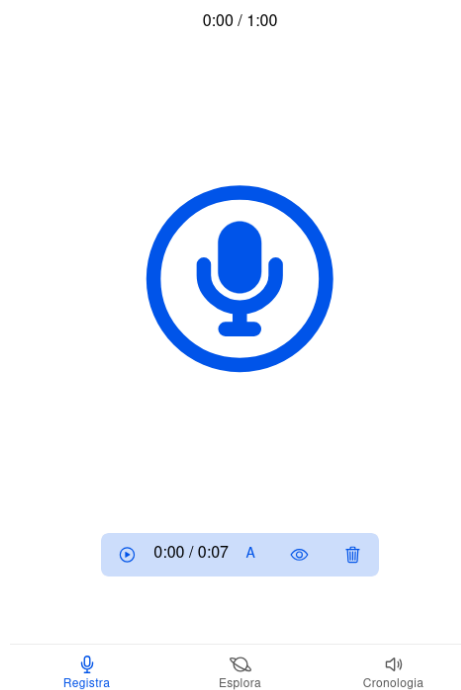


Figura 5: Schermata di registrazione audio con audio caricato





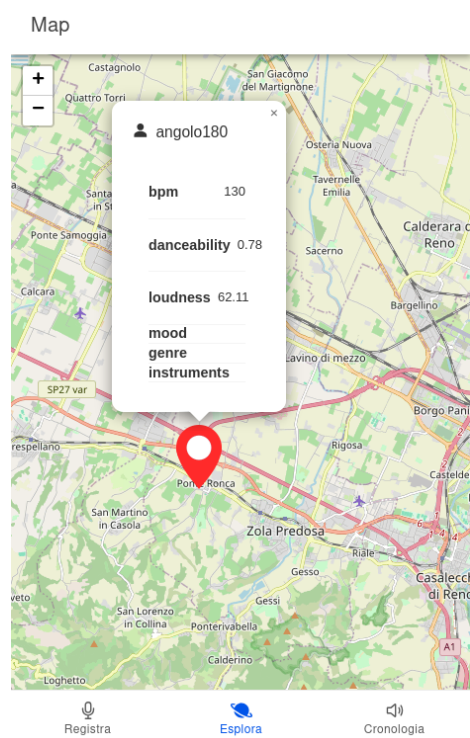


Figura 7: Schermata della mappa con le informazioni dell'audio

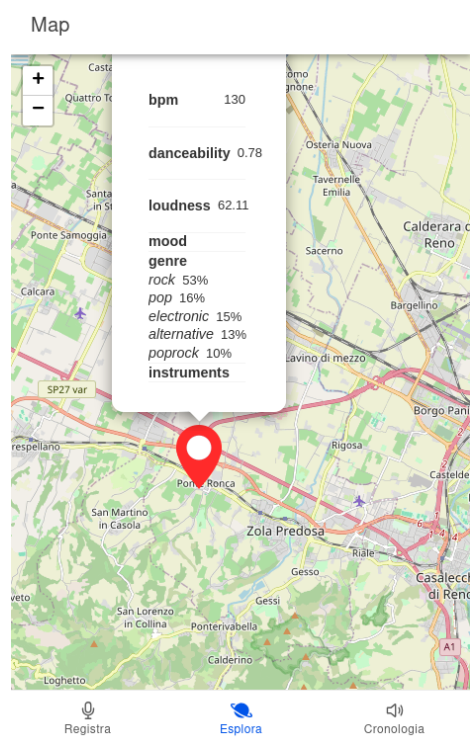


Figura 8: Schermata della mappa con il submenu dell'audio

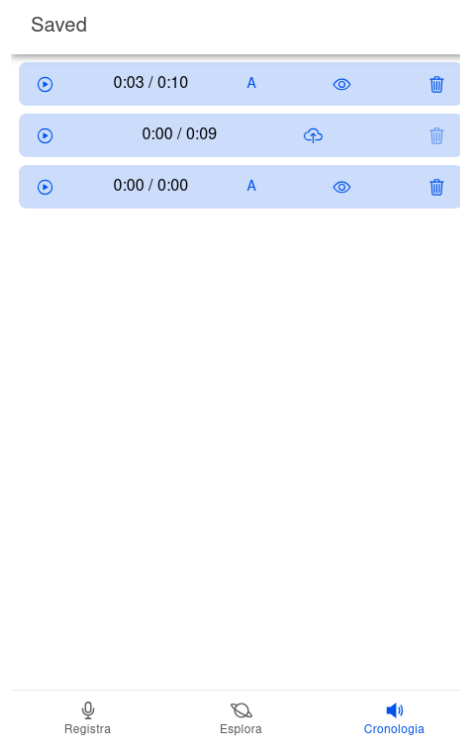


Figura 9: Schermata dei salvati

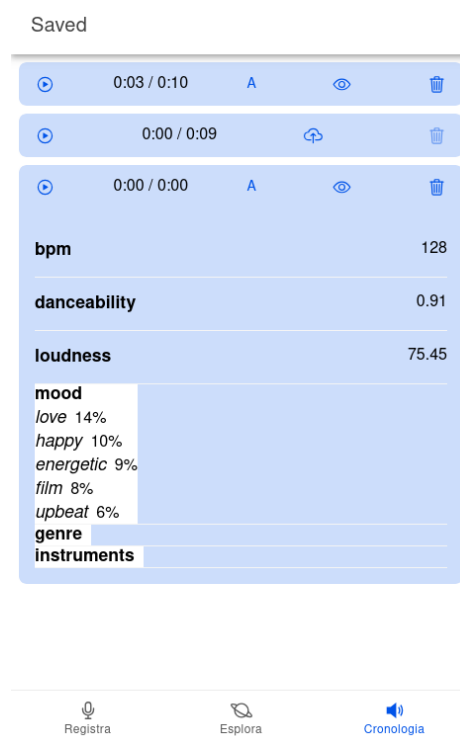


Figura 10: Schermata dei salvati con le informazioni