Implementazione

[Prototipo 1](#_Toc43591486)

[Architettura 1](#_Toc43591487)

[Postazione remota e sistema di visualizzazione 1](#_Toc43591488)

[Microfono integrato nel sistema di visualizzazione 2](#_Toc43591489)

[Sviluppo 2](#_Toc43591490)

[Arduino 2](#_Toc43591491)

[Node-red 3](#_Toc43591492)

[ThingSpeak 3](#_Toc43591493)

[Applicazione 3](#_Toc43591494)

# Prototipo

A Quiet Space nasce come soluzione innovativa per la risoluzione di problemi legati alla rumorosità negli uffici open space.

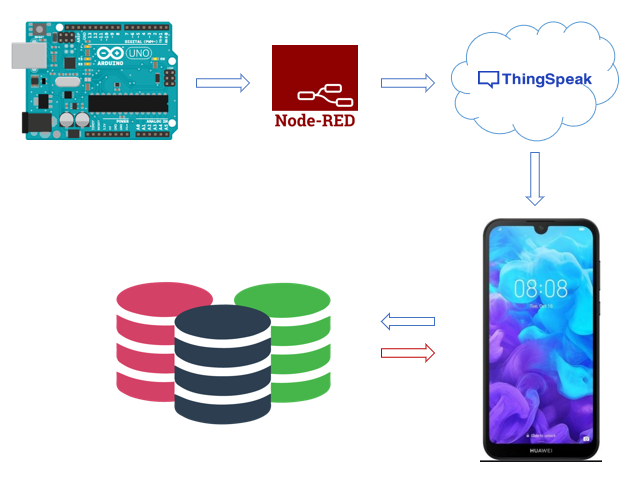
La soluzione adottata per lo sviluppo di questo servizio prevede l’utilizzo di piattaforme IoT che colleghino il microfono del dipendente al sistema informativo, rappresentato da un’applicazione o da un sito web, a seconda dell’esigenza aziendale.

## Architettura

Sono state individuate due principali architetture per “A Quiet Space”. Tutte hanno i seguenti punti in comune:

1. Un microfono per effettuare le registrazioni.
2. Un database cloud dove memorizzare i dati salvati.
3. Un sistema client dove elaborare i dati e visualizzarli.
4. Un server per memorizzare informazioni di runTime utili al corretto funzionamento del servizio.

### Postazione remota e sistema di visualizzazione

Questa architettura prevede l’utilizzo di una piattaforma remota di ascolto del livello sonoro, situata preferibilmente sulla scrivania del dipendente. Questo applicativo hardware ascolterà il suono circostante, catturerà il valore massimo in decibel e lo invierà al servizio cloud “ThingSpeak” mediante l’utilizzo di node-red o di un sensore wi-fi apposito.

Contemporaneamente l’applicazione (o sito web) leggeranno i dati sonori associati al dipendente e li visualizzeranno dopo averli adeguatamente elaborati.

### Microfono integrato nel sistema di visualizzazione

Questa architettura prevede l’utilizzo di un’unica piattaforma sia per l’ascolto della rumorosità che per la visualizzazione dei dati, ad esempio tramite applicazione smartphone oppure mediante microfono del pc e visualizzazione su apposito software o sito web.

Si mantiene ugualmente l’uso di ThingSpeak come database per i livelli sonori in quanto particolarmente adatto ad immagazzinare elevate quantità di dati.

## Sviluppo

Per lo sviluppo del prototipo è stata adottata la prima soluzione proposta: una piattaforma remota, implementata tramite un sistema Arduino ed un interfaccia client tramite un’applicazione smartphone. È stato inoltre implementato un database Altervista per la gestione dell’autenticazione e delle segnalazioni.

I linguaggi di programmazione utilizzati sono java per lo sviluppo dell’applicazione Android, PHP per l’interfaccia con il server e C++ per il codice implementato su Arduino.

Le piattaforme di sviluppo ed applicative utilizzate sono Android Studio, Altervista, PHPmyAdmin, ThingSpeak, Node-red ed Arduino.

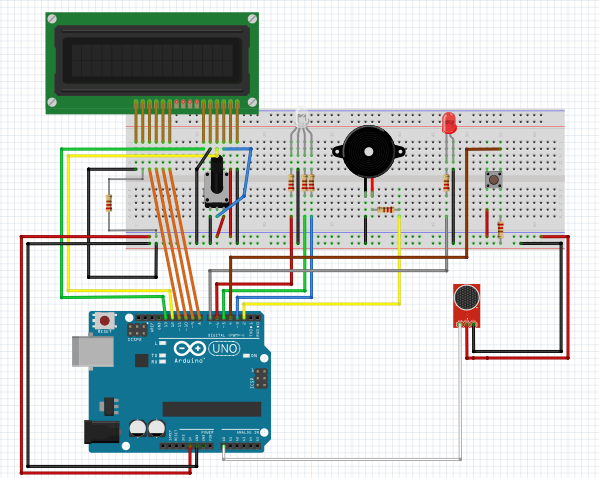
### Arduino

#### Schema di funzionamento

L’acquisizione del livello sonoro avviene tramite un sistema Arduino integrato con diverse componenti. L’obiettivo è quello di inviare il valore in decibel sul cloud e visualizzare una prima interfaccia per sensibilizzare l’utente in merito all’eventuale rumorosità eccessiva.

I componenti utilizzati sono i seguenti

* x1 scheda Eleego UNO
* x1 led RGB
* x1 led Arancio
* x1 Buzzer attivo
* x1 Bottone
* x1 Display LCD
* x4 resistenza 220 Ohm
* x1 resistenza 330 Ohm
* x1 resistenza 10k Ohm
* x1 microfono electret DFR0034



Come controllore viene utilizzata una scheda Eleego UNO, sostituto di quelle Arduino ma perfettamente compatibile con tutte le principali componenti. Il suono viene acquisito tramite un sensore DFR0034, ovvero un microfono di tipo electret. Questo sensore restituisce in output un valore di tensione tra 0 e 512 (osservato empiricamente) corrispondente ad un livello sonoro che oscilla tra circa 30db e 110db (rispetto ad un confronto con altri fonometri). Il valore di tensione viene trasmesso sul pin A0, elaborato dalla scheda centrale ed inviato al pc tramite porta seriale USB.

È presente un display LCD che visualizza i decibel letti dal sensore ed un avviso relativo a 3 livelli di rumorosità:

* “A Quiet Space!” per un rumore non fastidioso
* “Parla Piano!” Per indicare una situazione in cui si sta alzando un po' la voce, ma non ancora abbastanza per generare fastidio.
* “Abbassa la voce!” Per segnalare che il tono di voce risulta troppo elevato e deve essere ridotto.

Parallelamente al display LCD è presente anche un led RGB che segnala i 3 stati diversi tramite i colori verde, arancione e rosso. Per ogni ingresso del sensore sono state inserite delle resistenze da 220 Ohm in modo da ridurre il valore di tensione in ingresso. Per l’ingresso del colore rosso è stata utilizzata una resistenza da 330 Ohm in quanto solitamente nei sensori rgb il rosso predomina sugli altri colori ed è necessario riequilibrarlo affinché la codifica rgb risulti valida.

La restante parte del sistema è composta da un buzzer, attivato o disattivato da un bottone, al quale è associato un led per segnalarne lo stato. Il buzzer emette un suono per avvertire l’utente che si è superato il limite sonoro accettato per almeno 3 iterazioni (circa 5-6 secondi). Affinché il buzzer smetta di suonare è necessario ridurre il tono della voce entro limiti adeguati.

Tramite il bottone è possibile attivare o disattivare questa funzionalità in modo da evitare interruzioni fastidiose durante chiamate importanti. Lo stato Attivo/Disattivo è segnalato da un led arancione aggiuntivo.

Sia il led che il buzzer vengono posti in serie ad una resistenza di 220 Ohm mentre il bottone presenta un sistema PULL\_DOWN con un resistore di 10k Ohm per evitare false letture di tensione.

#### Codice

Il codice è sviluppato in linguaggio C++ in un ambiente di programmazione messo a disposizione da arduino che consente di interagire direttamente con la scheda e caricare immediatamente il codice funzionante, oltre a poter visualizzare eventuali output seriali.

DI seguito andremo ad analizzare le varie funzioni del codice, tralasciando la dichiarazione ed inizializzazione delle variabili e le associazioni dei pin ai segnali di input/output.

La funzione loop acquisisce il livello sonoro come tensione (oscillante tra 0 e 512 V) e ne memorizza il valore cumulato fino ad un totale di 10000 misurazioni, corrispondenti a circa 2 secondi. Viene quindi calcolata la media ed il valore risultante è utilizzato per il calcolo dei dB. La media viene effettuata per acquisire un valore verosimile del suono in quanto il segnale i valori in ingresso sono approssimabili a valori continui e sono presenti istanti a valore 0 seguiti da valori significativi a causa della discontinuità del livello sonoro in una conversazione.

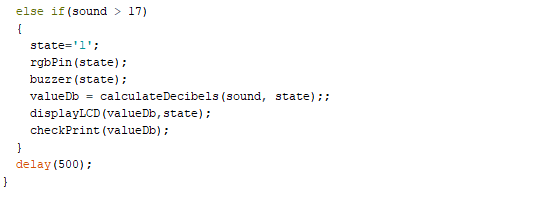
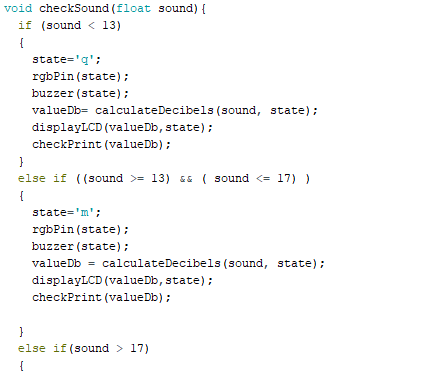
Viene quindi richiamata la funzione checkSound che si occupa di elaborare il segnale in volt appena acquisito.

Infine, viene effettuato un controllo sul bottone. Se questo è premuto (circuito chiuso => HIGH) viene attivato o disattivato il buzzer e segnalato tramite led. Lo stato del buzzer può essere cambiato una sola volta per lettura, quindi una volta ogni 2 secondi. Questa limitazione è imposta per garantire il corretto funzionamento del bottone ed acquisire un'unica variazione di stato per volta. Il bottone infatti non agisce di sua natura come uno switch, pertanto per simularne il comportamento è stato necessario discretizzare il segnale in input relativo al circuito del bottone, limitandone le acquisizioni.

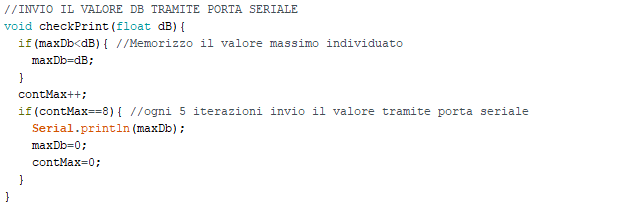


La funzione checkSound è il centro del programma di elaborazione. Al suo interno il segnale in volt viene assegnato ad uno di tre livelli, corrispondenti ad una rumorosità bassa, moderatamente alta oppure eccessiva. Le tre soglie sono state elaborate partendo da un progetto di misura del suono già esistente ed aggiustate tramite prove empiriche al microfono a nostra disposizione. La stessa tecnica è stata utilizzata per la trasformazione del valore in volt in Decibel, che utilizza tre diverse funzioni logaritmiche in funzione dei tre livelli possibili.

Per ogni livello viene memorizzato uno stato e questo viene passato ad altre funzioni che gestiscono il comportamento delle altre componenti.



Infine, il valore del suono in decibel, tramite la funzione checkPrint, viene confrontato con le 7 successive acquisizioni ed il valore massimo verrà inviato tramite porta USB seriale. Sarà quindi inviato un valore ogni 15 secondi circa, rispettando i vincoli imposti da ThingSpeak.

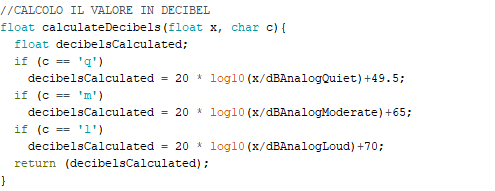


La funzione calculateDecibel, richiamata in checkSound, si occupa della trasformazione del valore del suono da Volt a Decibel. Viene applicata una funzione logaritmica che prende in considerazione il livello sonoro.

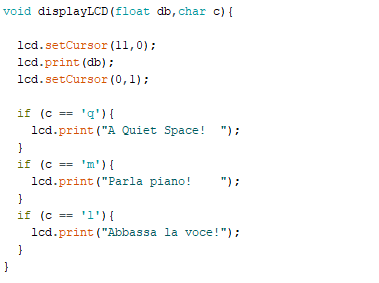
Il valore dbAnalog… è stimato empiricamente tramite formula inversa e varia in funzione del microfono utilizzato. I valori da noi ottenuti sono:

* dBAnalogQuiet=10, simulando un suono pari a 48dB
* dBAnalogModerate=12, simulando un suono pari a 62dB
* dBAnalogLoud=17 simulando un suono pari a 73dB

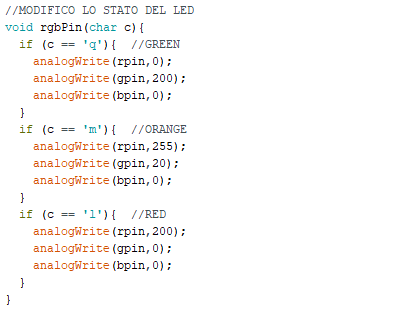
Questi valori sono inoltre stati utilizzati come soglie per la distinzione tra i tre livelli sonori possibili.



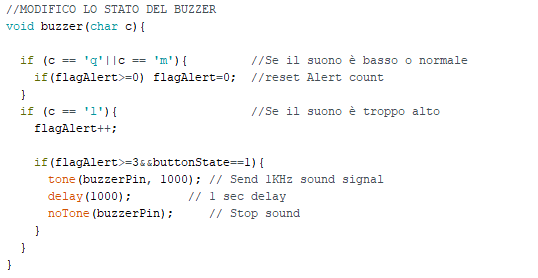
La funzione displayLCD, ottenuto il valore in Decibel, lo stampa a video sul display insieme ad un avviso basato sullo stato del suono.



La funzione LedRGB, similmente al display, invia un segnale diverso in funzione dello stato, permettendo l’emissione di un segnale visivo verde, arancione o rosso.



Per concludere, la funzione buzzer gestisce il processo di attivazione/disattivazione del buzzer qualora si superi la soglia massima per 3 acquisizioni consecutive.

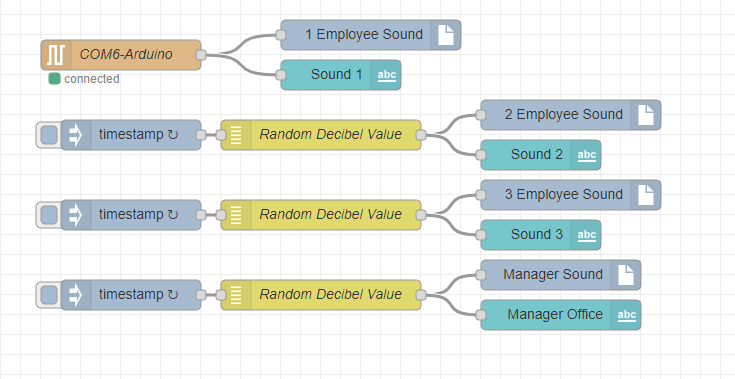


### Node-red

L’interfaccia di node-red è utilizzata principalmente come modulo wi-fi per la piattaforma Arduino. Raccoglie i dati in ingresso dalla porta usb del pc contenenti i valori in decibel e li invia a ThingSpeak tramite protocollo http.

Per simulare i vari dipendenti all’interno sono stati utilizzati dei timestamp che generano un segnale ogni 16 secondi. Il segnale attiva a sua volta un generatore di numeri casuali, impostato tra 45 e 75 dB che invia poi i dati a ThingSpeak. In questo modo abbiamo simulato la presenza di ulteriori 2 dipendenti in uno stesso ufficio più 1 manager, saturando il numero di canali messi a disposizione da ThingSpeak.

È inoltre presente una piccola dashboard per la visualizzazione dei decibel trasmessi, utile in fase di debug del sistema.

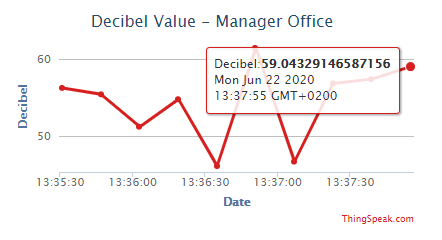


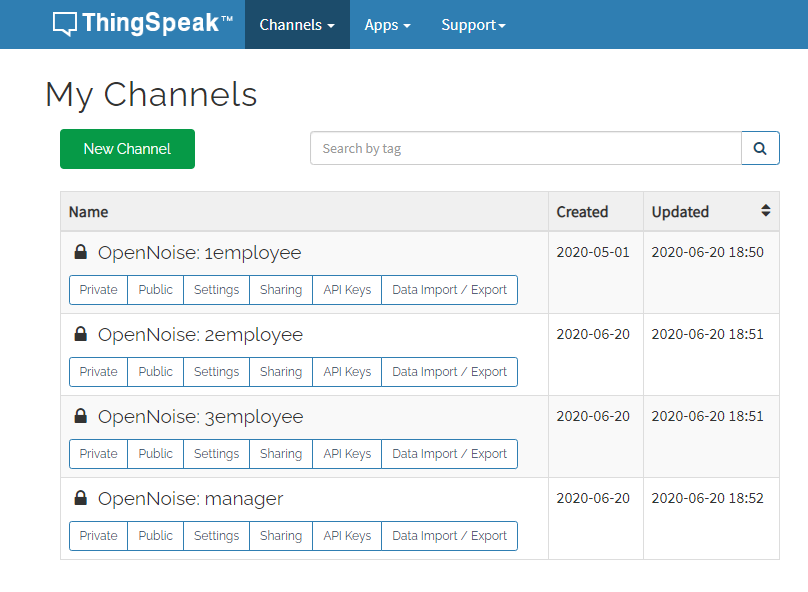
### ThingSpeak

ThingSpeak è una piattaforma cloud che consente la memorizzazione di dati provenienti da sensori applicati nei vari sistemi. Ne permette inoltre l’elaborazione direttamente in cloud grazie ad un’integrazione con Matlab. La versione gratuita ha purtroppo delle limitazioni: consente un massimo di 4 canali contemporaneamente ed ogni canale può elaborare una richiesta http ogni 15 secondi.

Nonostante queste limitazioni, è stato deciso di utilizzarlo a fini prototipali in quanto consente la memorizzazione di un elevato quantitativo di dati, come richiesto dal nostro sistema, il quale ha la necessità di memorizzare un numero elevato di misurazioni su cui effettuare diverse analisi.

Nel dettaglio abbiamo predisposto 4 canali. 3 canali sono associati a 3 dipendenti dell’ufficio marketing mentre il 4 è relativo al manager dell’ufficio che si trova in un’altra area.



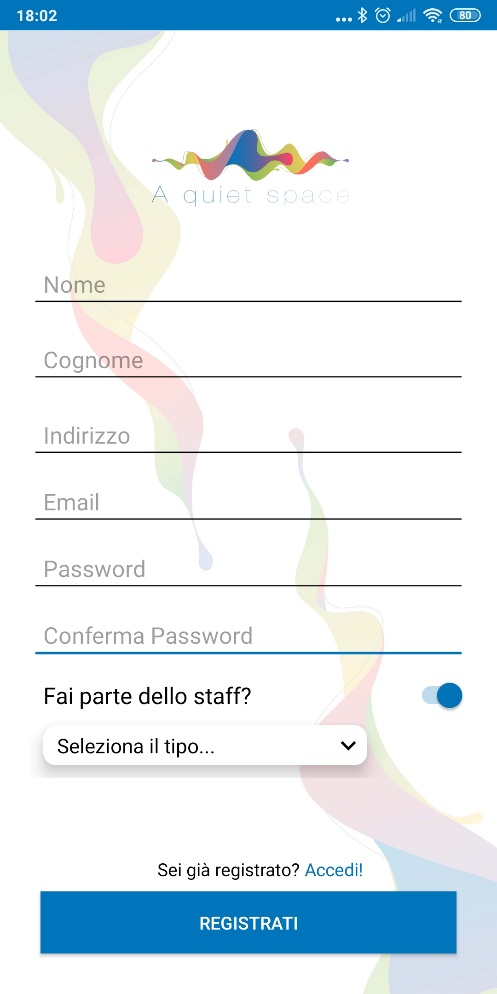
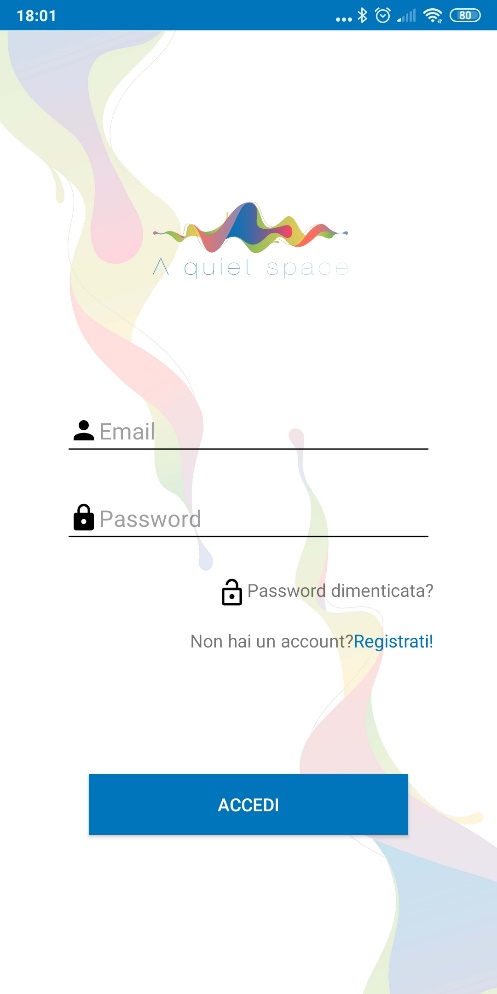


### Applicazione

l’applicazione, “A Quiet Space” permette di monitorare il livello sonoro nell'ambiente circostante, segnalando condizioni di rumorosità eccessiva.

L'app consente di registrarsi tramite un'apposita procedura, inserendo i propri dati anagrafici, l’e-mail lavorativa e la password.

è possibile inoltre indicare lo status di membro staff, indicando la figura corrispondente. Il nuovo account verrà quindi memorizzato sul server ed inizierà la procedura di validazione tramite e-mail. Una volta validato l’account sarà possibile effettuare il login all’applicazione tramite l’apposita pagina.

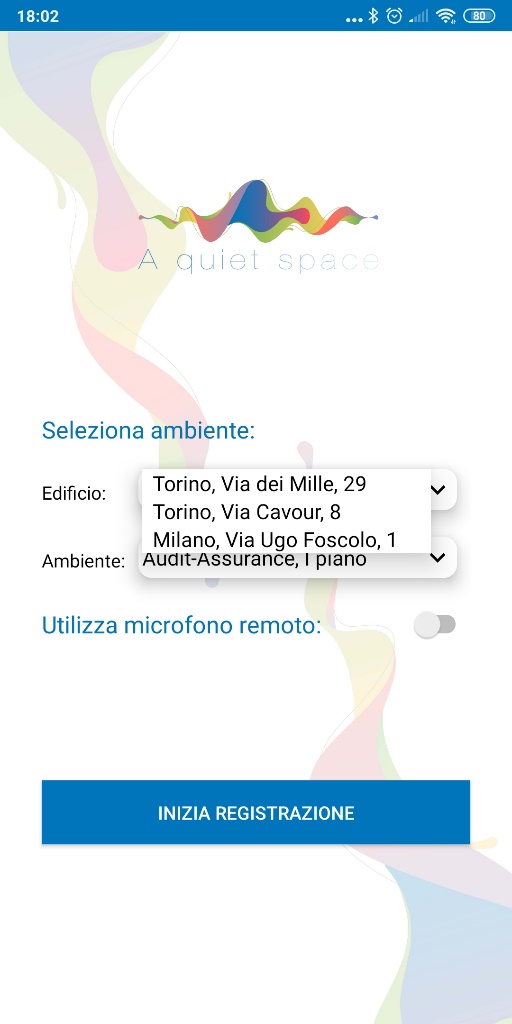
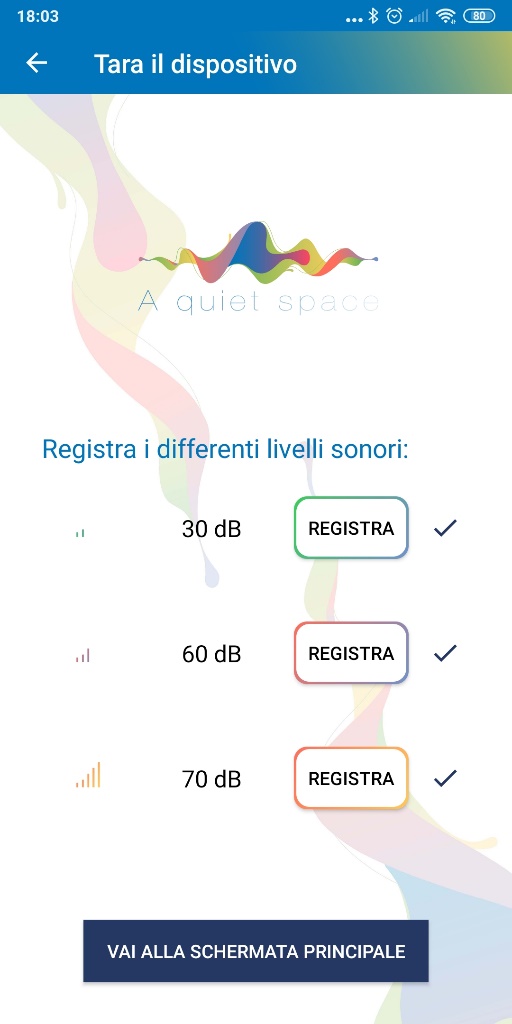
 

Effettuando il login si viene reindirizzati ad una pagina in cui si seleziona la sede e l’ufficio Open Space in cui si andrà a lavorare durante la giornata.

È inoltre necessario indicare se il dispositivo di registrazione utilizzato è un microfono remoto oppure quello integrato nel dispositivo.

Il microfono remoto è rappresentato da una piattaforma arduino, situata sulla postazione di lavoro che cattura il livello sonoro circostante. Affinché lo smartphone venga associato correttamente al dispositivo remoto, è necessario inserire due stringhe corrispondenti agli identificativi del canale di comunicazione utilizzato dal sistema.

Di default viene invece utilizzato il microfono integrato all’interno dello smartphone. Questa necessità di una iniziale taratura per garantire una lettura affidabile del suono.

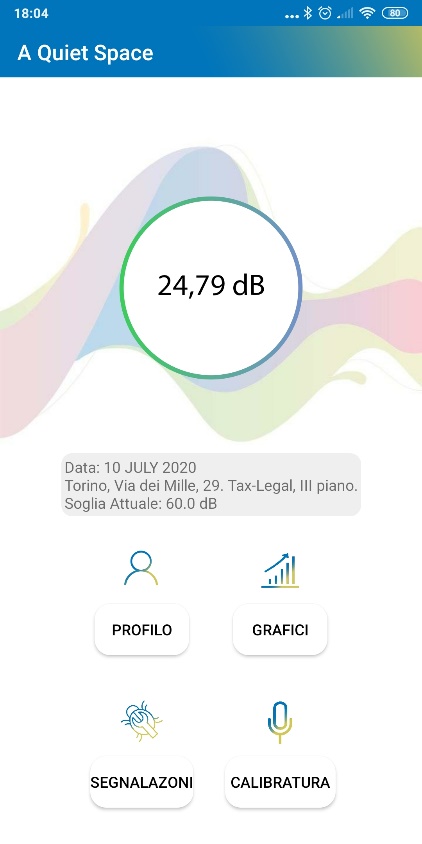
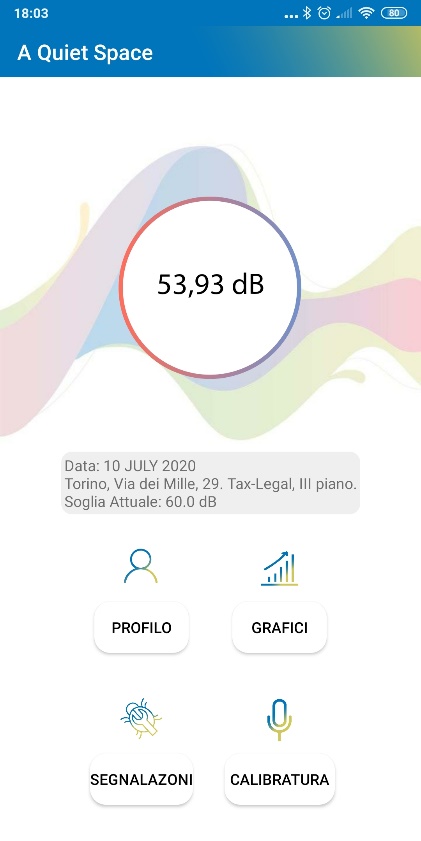
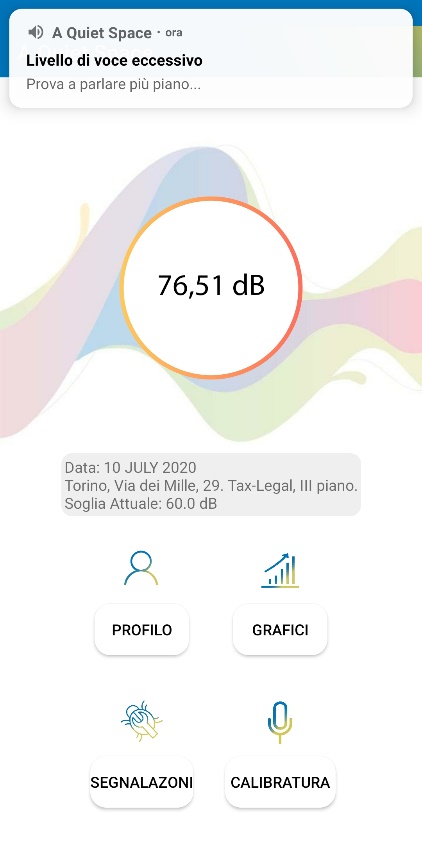
Vengono ascoltati per circa 5 secondi 3 diversi livelli sonori corrispondenti a 30, 60 e 70 decibel. I suoni di test saranno individuati utilizzando rumori costanti misurati tramite dispositivi ad alta precisione e successivamente forniti durante la procedura di convalida dell’account.

Dopo aver calibrato il dispositivo si viene reindirizzati alla pagina principale. Qui sono riepilogate le informazioni sulla stanza e sulla soglia impostata. È inoltre presente una sezione per navigare all’interno delle 4 differenti sezioni esistenti ed una in cui è visualizzato il livello di rumorosità rilevato durante l’ultima cattura del suono.

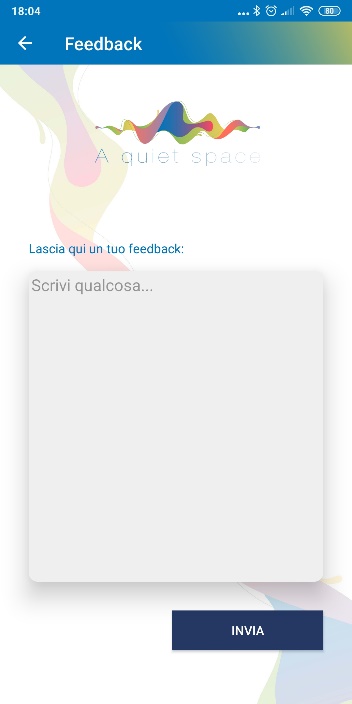
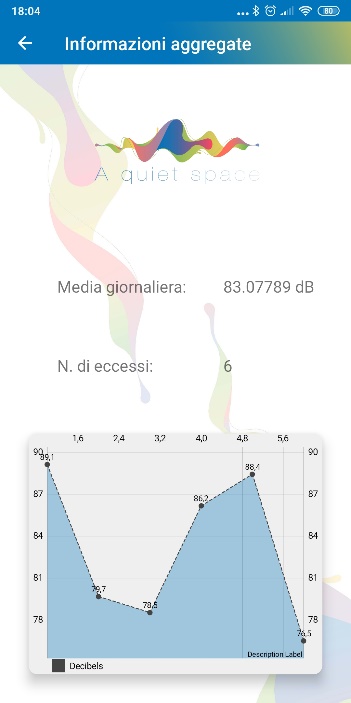
Un’operazione di cattura del suono dura circa 15 secondi. Il livello sonoro viene monitorato lungo questo intervallo e ne viene estratto il valore medio, il quale viene convertito in decibel tramite un’apposita funzione a 3 livelli basata sulla taratura effettuata.

In base al livello sonoro rilevato verrà visualizzata una differente interfaccia grafica ed in caso di rumorosità eccessiva (3° livello), un sistema di notifiche avvertirà l’utente di abbassare il tono di voce. Il livello di rumorosità eccessiva è individuato tramite una soglia impostata dallo staff tecnico per ogni ufficio open space, in funzione delle dimensioni e della tipologia.

La notifica avviene mediante avviso visivo per i primi due eccessi consecutivi, dal terzo è aggiunto un segnale audio, disattivabile dall’utente.



Dalla pagina principale è possibile navigare in 4 sezioni differenti:

* **Profilo:** consente di cambiare la password, visualizzare le informazioni relative alla privacy, disattivare l’avviso sonoro ed effettuare il logout dall’applicazione
* **Grafici:** visualizza le statistiche relative agli eccessi rilevati. È mostrata una media giornaliera del livello eccessivo, il numero di eccessi rilevati ed una rappresentazione grafica nel tempo degli eccessi.
* **Segnalazioni:** Permette di segnalare eventuali problemi o rilasciare feedback in merito all’applicazione.
* **Calibratura:** Consente di effettuare nuovamente la procedura di calibrazione del dispositivo nel caso si stia usando un microfono integrato.