



# HOW IT WORKS



Multimedia App & Internet of  
Things Project

Introduzione

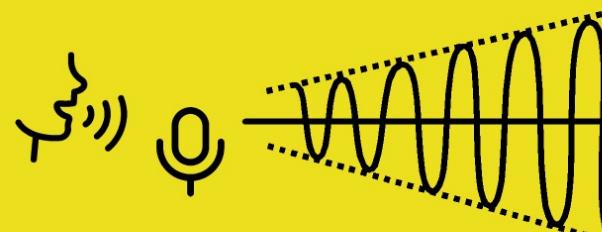
Demo

Progettazione

Conclusione

## Idea di progetto

Abbiamo realizzato un'applicazione per testare lo stato mnesico di un soggetto e quindi valutare la presenza di Alzheimer.



I.M.C.T

## Semplificazione della realtà

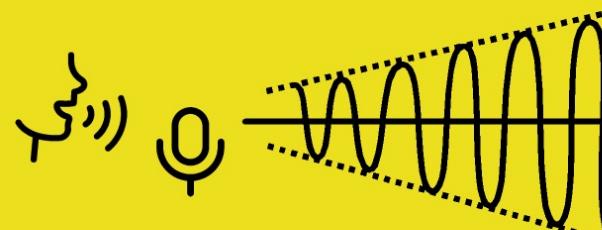
... abbiamo deciso quindi di estrapolare una parte di un test utilizzato per valutare lo stato mentale di un soggetto chiamato I.M.C.T (Information Memory Concentration Test), il quale pone delle domande alle quali il paziente dovrà rispondere attraverso l'uso della voce...

L'applicazione riconoscerà attraverso dei modelli di Machine Learning le risposte e ritornerà se queste ultime sono corrette o meno e la lingua nella quale sono state pronunciate...



## Idea di progetto

Abbiamo realizzato un'applicazione per testare lo stato mnesico di un soggetto e quindi valutare la presenza di Alzheimer.



I.M.C.T



# HOW IT WORKS



Multimedia App & Internet of  
Things Project

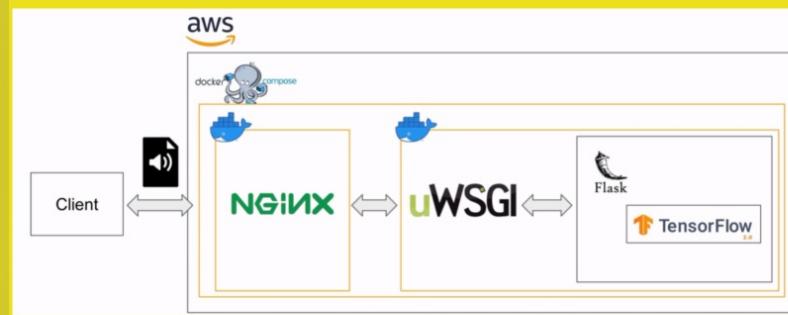
Introduzione

Demo

Progettazione

Conclusione

# Architettura



Machine  
Learning

Android

Collegamento

## **Architettura della Web API**

Per iniziare abbiamo sviluppato una Web API tramite Python e Flask la quale ci ha permesso di interrogare i modelli di Machine Learning che abbiamo allenato per eseguire Word e Language Recognition

## **Analisi dei file audio**

## Spettrogrammi

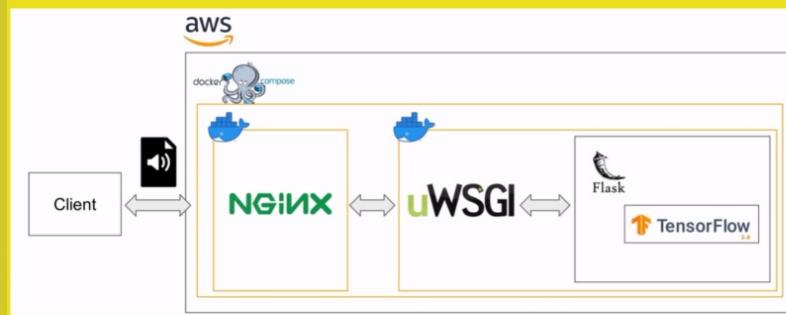
Per allenare i modelli di Machine learning abbiamo utilizzato una metodologia di **apprendimento supervisionato** nella quale i file audio venivano elaborati e rappresentati attraverso una rappresentazione dello spettro di potenza a breve termine di un suono chiamata **MFCC** (Mel Frequency Cepstral Coefficient), la quale permette di identificare le componenti del segnale audio che sono utili per identificare il contenuto linguistico e scartare tutte le altre cose che trasportano informazioni come rumore di fondo, emozioni ecc.

## **Architettura della Web API**

Per iniziare abbiamo sviluppato una Web API tramite Python e Flask la quale ci ha permesso di interrogare i modelli di Machine Learning che abbiamo allenato per eseguire Word e Language Recognition

## **Analisi dei file audio**

# Architettura



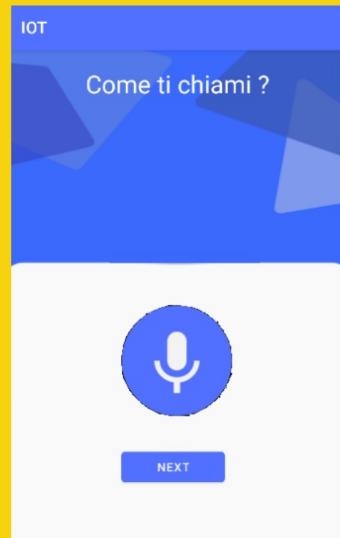
Machine  
Learning

Android

Collegamento

## Progettazione del Quiz

Abbiamo estrapolato una ventina di domande dal test originale, alle quali l'utente potrà rispondere iniziando e terminando la registrazione premendo il tasto



## Report

# Progettazione Report

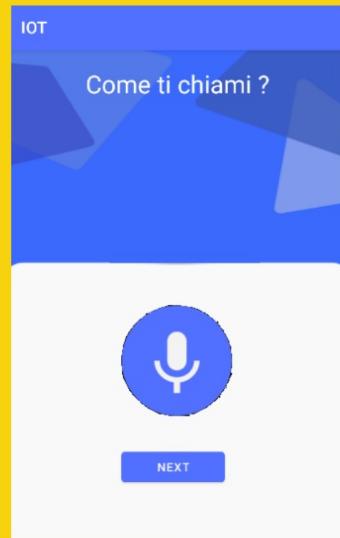
Al termine del quiz volevamo mostrare i risultati attraverso una semplice schermata nella quale mostrare il numero di risposte corrette, sbagliate ed il numero di risposte date nelle diverse lingue: italiano, inglese e francese...

Riportando in conclusione un messaggio a seconda dei risultati ottenuti.



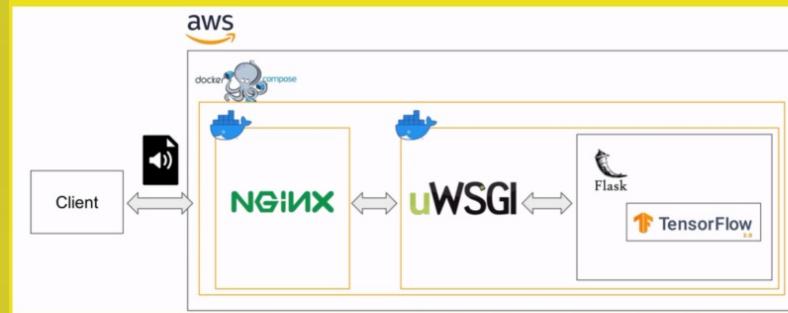
## Progettazione del Quiz

Abbiamo estrapolato una ventina di domande dal test originale, alle quali l'utente potrà rispondere iniziando e terminando la registrazione premendo il tasto



## Report

# Architettura



Machine  
Learning

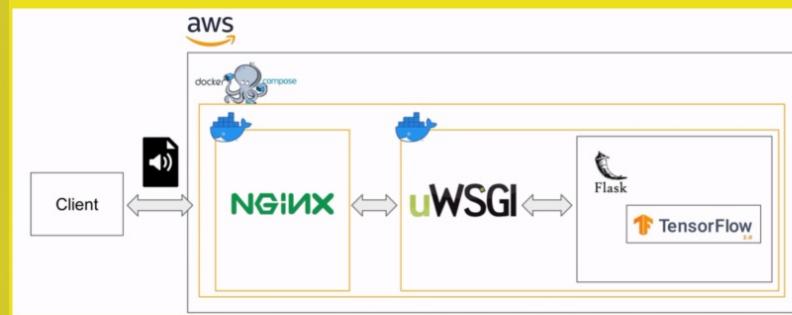
Android

Collegamento

## **Connessione al Server AWS**

Per poter effettuare le richiesta al server contenente i modelli di machine learning abbiamo creato una classe apposita (`HttpPostAsyncTask`), la quale permette di eseguire richieste HTTP di tipo POST in background

# Architettura



Machine  
Learning

Android

Collegamento



# HOW IT WORKS



Multimedia App & Internet of  
Things Project

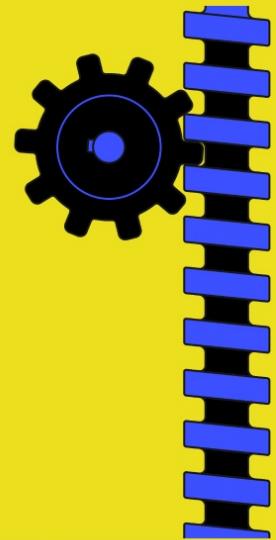
Introduzione

Demo

Progettazione

Conclusione

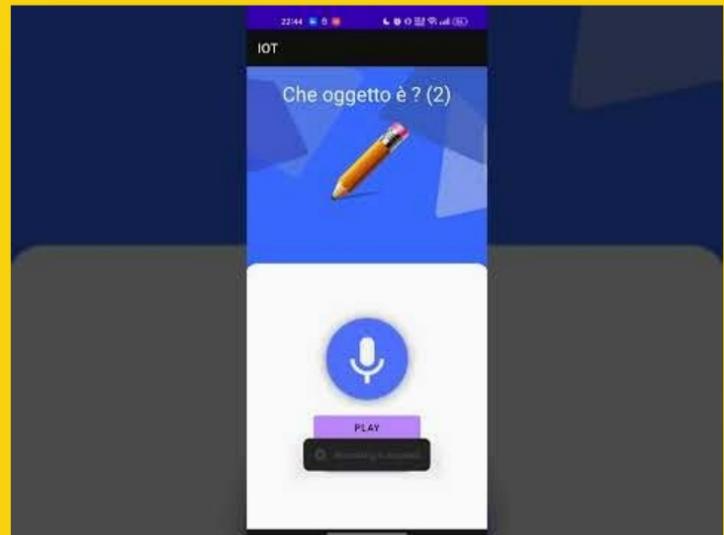
## Funzionamento



- 1 Creazione profilo
- 2 Registrazione risposta
- 3 Analisi audio
- 4 Risposta dal modello
- 5 Report finale

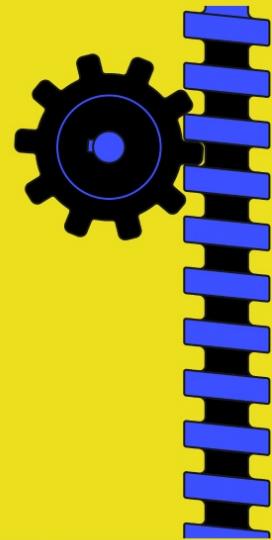


## Video demo





## Funzionamento



- 1 Creazione profilo
- 2 Registrazione risposta
- 3 Analisi audio
- 4 Risposta dal modello
- 5 Report finale





# HOW IT WORKS



Multimedia App & Internet of  
Things Project

Introduzione

Demo

Progettazione

Conclusione

## Conclusioni

Ovviamente il nostro test non è da ritenere attendibile poiché non abbiamo potuto allenare i modelli di machine learning al meglio e quindi spesso non forniscono risultati accurati...

Ma il nostro obiettivo era fornire un test iniziale che potesse permette un'analisi preventiva del problema...



Fine...

## Gruppo 12



**Celeste Bazzi** - 4840738

- Progettazione, Design e Report

**Davide Miggiano** - 4840761

- Machine learning, Web API

**Filippo Manini** - 4798004

- Quiz, Registrazione audio e connessione

## Conclusioni

Ovviamente il nostro test non è da ritenere attendibile poiché non abbiamo potuto allenare i modelli di machine learning al meglio e quindi spesso non forniscono risultati accurati...

Ma il nostro obiettivo era fornire un test iniziale che potesse permette un'analisi preventiva del problema...



Fine...

## Tecnologie usate



- 1 Android
- 2 Python/Flask
- 3 TensorFlow
- 4 Docker
- 5 uWSGI
- 6 NGINX

## Conclusioni

Ovviamente il nostro test non è da ritenere attendibile poiché non abbiamo potuto allenare i modelli di machine learning al meglio e quindi spesso non forniscono risultati accurati...

Ma il nostro obiettivo era fornire un test iniziale che potesse permette un'analisi preventiva del problema...



Fine...

**Grazie per l'attenzione**



## Conclusioni

Ovviamente il nostro test non è da ritenere attendibile poiché non abbiamo potuto allenare i modelli di machine learning al meglio e quindi spesso non forniscono risultati accurati...

Ma il nostro obiettivo era fornire un test iniziale che potesse permette un'analisi preventiva del problema...



Fine...



# HOW IT WORKS



Multimedia App & Internet of  
Things Project

Introduzione

Demo

Progettazione

Conclusione