



UNIVERSITÀ  
DEGLI STUDI  
FIRENZE

Scuola di Scienze Matematiche, Fisiche e Naturali  
Corso di Laurea Magistrale in Informatica

Advanced Tools and Techniques for Software Development

## REHEARSAL ROOM SCHEDULE WEB SERVICE

FRANCESCO TERROSI

6326113

Anno Accademico 2018-2019

---

## INDICE

---

1	Introduzione al Software	3
	1.0.1	Descrizione e Vincoli     3
	1.0.2	Panoramica del Software     3
2	Conclusioni	5



---

## INTRODUZIONE AL SOFTWARE

---

### 1.0.1 *Descrizione e Vincoli*

Il software sviluppato implementa un web service per la gestione delle prenotazioni di una sala prove.











Lo sviluppo del software è iniziato definendo dei requisiti sull'orario della sala e sulla definizione delle richieste di prenotazione valide. In particolare:

- La sala prove in questione offre 3 sale, prenotabili per turni di 2 ore e 30 minuti
- Ciascuna sala è prenotabile per qualunque orario (il minutaggio non deve necessariamente essere un multiplo di 30, la sala prove rimane aperta h24)
- È possibile effettuare una prenotazione fino a 5 minuti prima dell'orario specificato
- Tutte le prenotazioni richiedenti un'orario  $o$ ,  $o \leq \text{ora attuale} + 5 \text{ minuti}$  vengono rifiutate
- Soltanto gli utenti registrati nel sistema possono effettuare le prenotazioni
- Le prenotazioni vengono associate in maniera univoca all'username scelto in fase di registrazione
- Non possono esistere due utenti con lo stesso username

### 1.0.2 *Panoramica del Software*

Il software è stato implementato seguendo una filosofia di sviluppo modulare. Possiamo dividere i vari pacchetti in pacchetti di **utilità** (model,

exceptions, configurations. . . ) e pacchetti di **servizio** (services, repository, web)

- 
- ▼  rehearsal-room [boot] [rehearsal-room-project master]
    - ▼  src/main/java
      - ▶  org.unifi.ft.rehearsal
      - ▶  org.unifi.ft.rehearsal.annotations
      - ▶  org.unifi.ft.rehearsal.configurations
      - ▶  org.unifi.ft.rehearsal.exceptions
      - ▶  org.unifi.ft.rehearsal.model
      - ▶  org.unifi.ft.rehearsal.repository.mongo
      - ▶  org.unifi.ft.rehearsal.services
      - ▶  org.unifi.ft.rehearsal.web

---

## CONCLUSIONI

---

Lo scopo di questo progetto era quello di condurre un'analisi più approfondita sulla possibilità di riconoscere gli utenti e le frasi da loro pronunciate durante una conversazione Skype e di riuscire a distinguere fra videochiamate, conferenze e chiamate fra due utenti.

Purtroppo i dati riportati ci dimostrano come alcuni di questi obiettivi siano impossibili da raggiungere, in particolare non è stato possibile:

1. Identificare elementi esterni alla conversazione
2. Identificare tratti distintivi nella parlata di un utente
3. Stabilire la lingua della conversazione (a meno di utilizzo di software come Skypegrep)

L'impossibilità di questi 3 punti (salvo casi specifici per il punto (1)) è da attribuire quasi completamente al codec a bitrate variabile utilizzato da Skype per la cattura dei messaggi. In questo modo è impossibile riconoscere uno specifico utente, nè è possibile catturare rumori esterni. Rimane tuttavia possibile identificare *specifiche* frasi all'interno di una conversazione.

In ogni caso sono stati ottenuti risultati interessanti per quanto riguarda:

1. La possibilità di distinguere se è in corso una chiamata, una videochiamata o una conferenza
2. La possibilità di capire se vi sono fonti di rumore *costante* esterno alla conversazione
3. A seguito di un adeguato train-set, la possibilità nella maggior parte dei casi identificare le frasi o le parole pronunciate

Per quanto riguarda il terzo punto è importante ribadire che non è esattamente la frase ad essere riconosciuta (ovvero non è possibile, analizzando i pacchetti, capire *quale* frase sia stata pronunciata) ma è possibile osservare la presenza di determinate sequenze di pacchetti, riconducibili a frasi specifiche.

Nel condurre gli esperimenti, come già è stato detto all'inizio di questo documento, sono stati applicati dei filtri con *Wireshark* in modo tale da ridurre al minimo le fonti di incertezza sui dati. È utile notare tuttavia che:

- È possibile che un'elevata latenza di rete disturbi la qualità delle osservazioni
- Nel monitoraggio delle conferenze non è sempre possibile stabilire il destinatario dei pacchetti, costringendo ad applicare meno filtri e quindi a catturare anche pacchetti non relativi al traffico Skype
- Nell'analisi delle videochiamate e delle conferenze sono stati stabiliti dei bound arbitrari per la cattura dei pacchetti audio. Per quanto i risultati fossero in linea con le precedenti osservazioni non è da escludere che alcuni pacchetti siano stati involontariamente esclusi