

# PROGETTO TECNOLOGIE INTERNET



*Andrea Bertogalli – Gabriele Fagnoni*

*(2021-2022)*

## SOMMARIO

|                             |   |
|-----------------------------|---|
| Introduzione .....          | 3 |
| Login.....                  | 3 |
| Signup .....                | 3 |
| Sviluppo del progetto ..... | 6 |
| Tecnologie utilizzate ..... | 7 |
| Backend .....               | 7 |
| Frontend.....               | 7 |

## INTRODUZIONE

Come progetto per il corso di Tecnologie Internet si è voluta sviluppare una piattaforma di video chat sulla quale gli utenti iscritti possano comunicare tra loro.

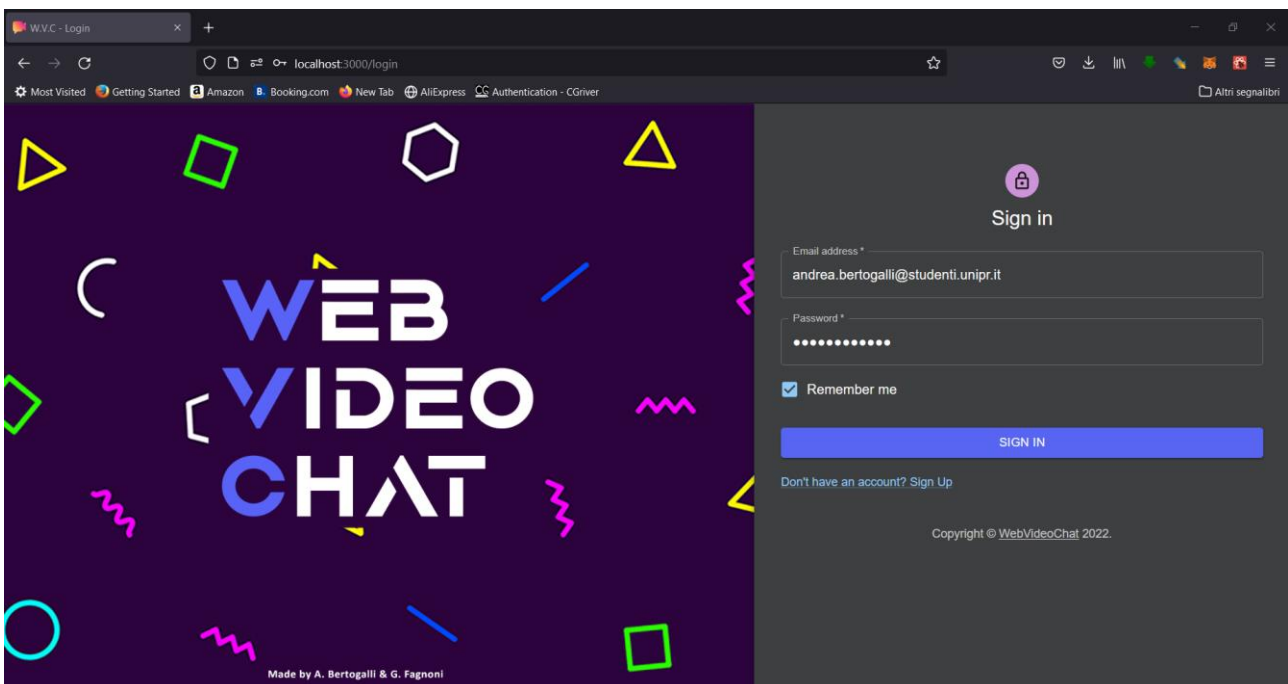
*Di seguito sono riportati i requisiti utente quindi cosa un utente può fare all'interno della piattaforma.*

La piattaforma prevede tre principali sezioni:

- Login
- Signup
- Dashboard

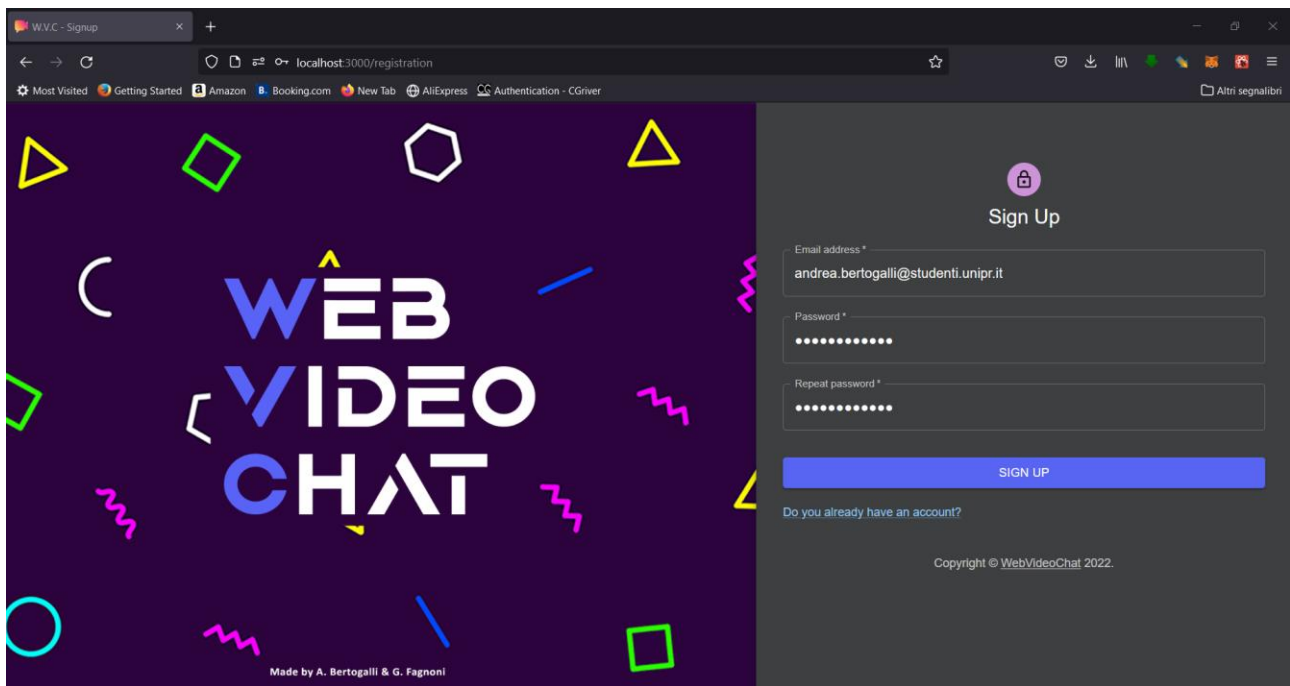
## LOGIN

Nella parte dedicata al login un utente iscritto alla piattaforma può fornire, attraverso un apposito form le sue credenziali che sono composte da e-mail e password, l'apposito bottone di login permetterà di inviare una richiesta di autenticazione al server che confermerà o meno la possibilità di accesso al sistema. In caso le credenziali siano esatte l'utente verrà reindirizzato presso la dashboard mentre se sono errate verrà comunicato l'esito negativo tramite un apposito messaggio. Il form di login prevede inoltre la possibilità di rimanere connessi, in questo caso, fino a che l'utente non sceglierà il contrario effettuando il logout la piattaforma autenticherà il dispositivo e quindi l'utente in automatico. Dalla pagina di login è inoltre possibile spostarsi alla pagina di registrazione tramite l'apposito link.



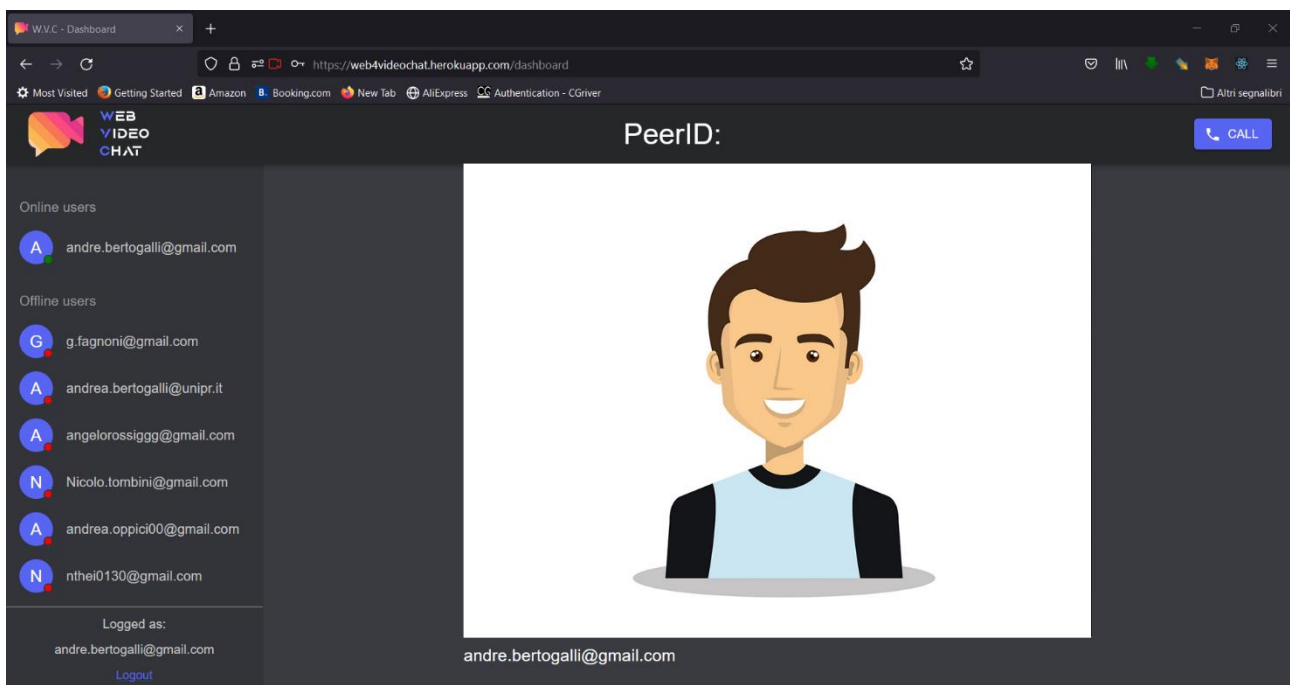
## SIGNUP

Questa pagina permette ad utenti non registrati di registrarsi alla piattaforma, anche in questa sezione è presente un form che permette di inserire le credenziali di un nuovo utente solo se non è già presente un utente con lo stesso indirizzo e-mail, anche in questa pagina l'invio verso il server è reso possibile tramite un apposito bottone di Signup. Una volta confermata la registrazione dal server l'utente procederà presso la pagina di login altrimenti verrà mostrato un messaggio di errore.

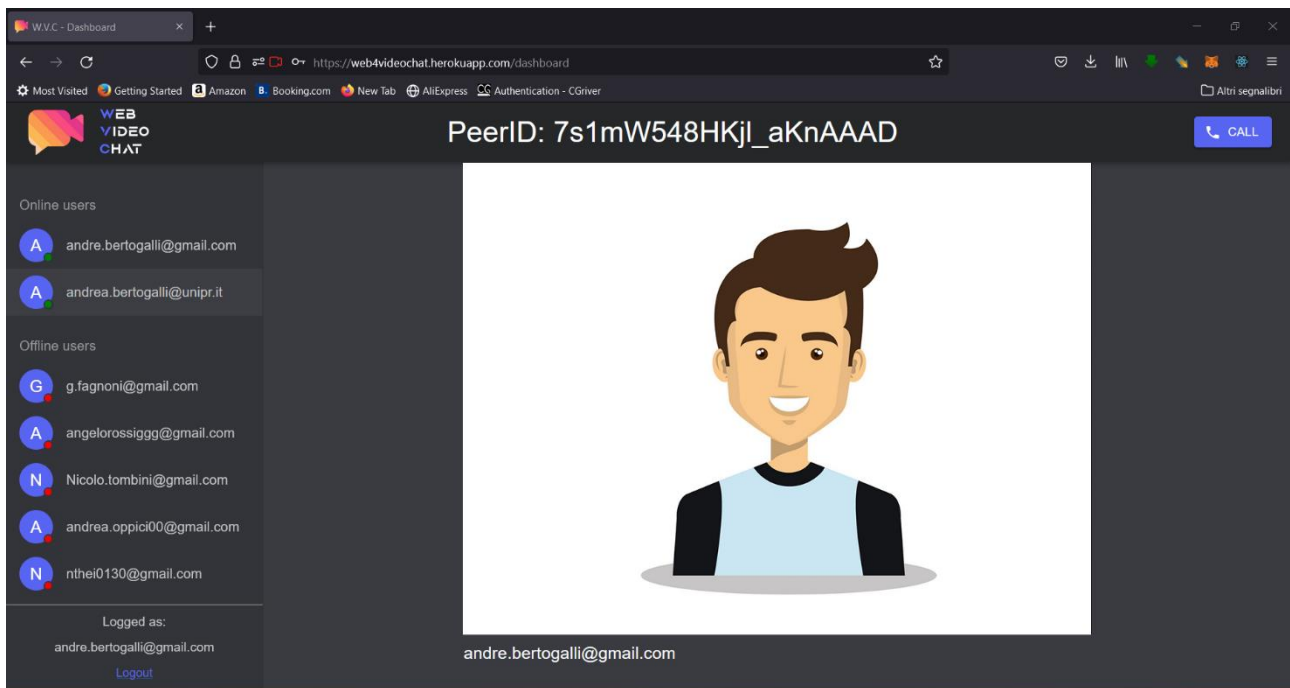


## DASHBOARD

Dopo aver fatto l'accesso con successo, all'utente viene chiesto dal browser se può esser dato l'accesso al microfono e alla videocamera per poter fare una normale chiamata con altri utenti. In questo caso si ritrova subito sulla pagina della dashboard, in cui può vedere al centro la ripresa di sé stesso ed un menù alla sua sinistra.

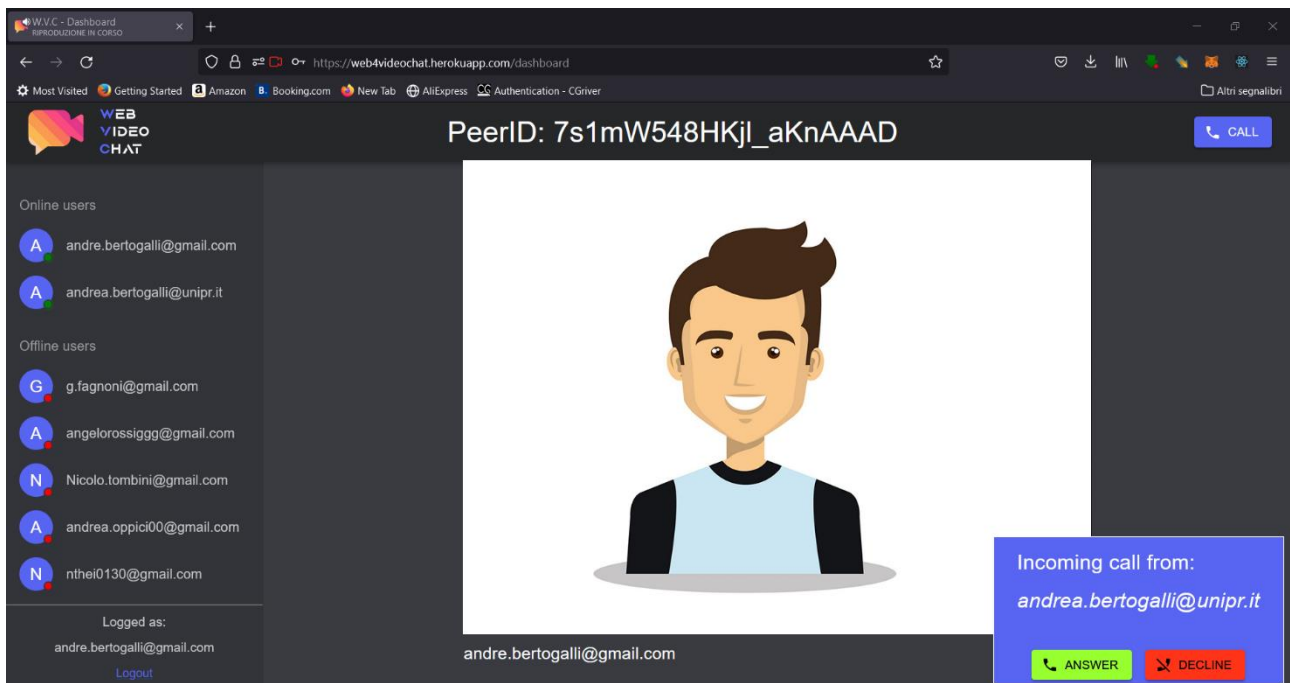


Il menù mostra gli utenti sia offline che online selezionabili per fare una videochiamata. Sopra la propria videoripresa viene inoltre mostrato l'identificativo del socket dell'utente selezionato.

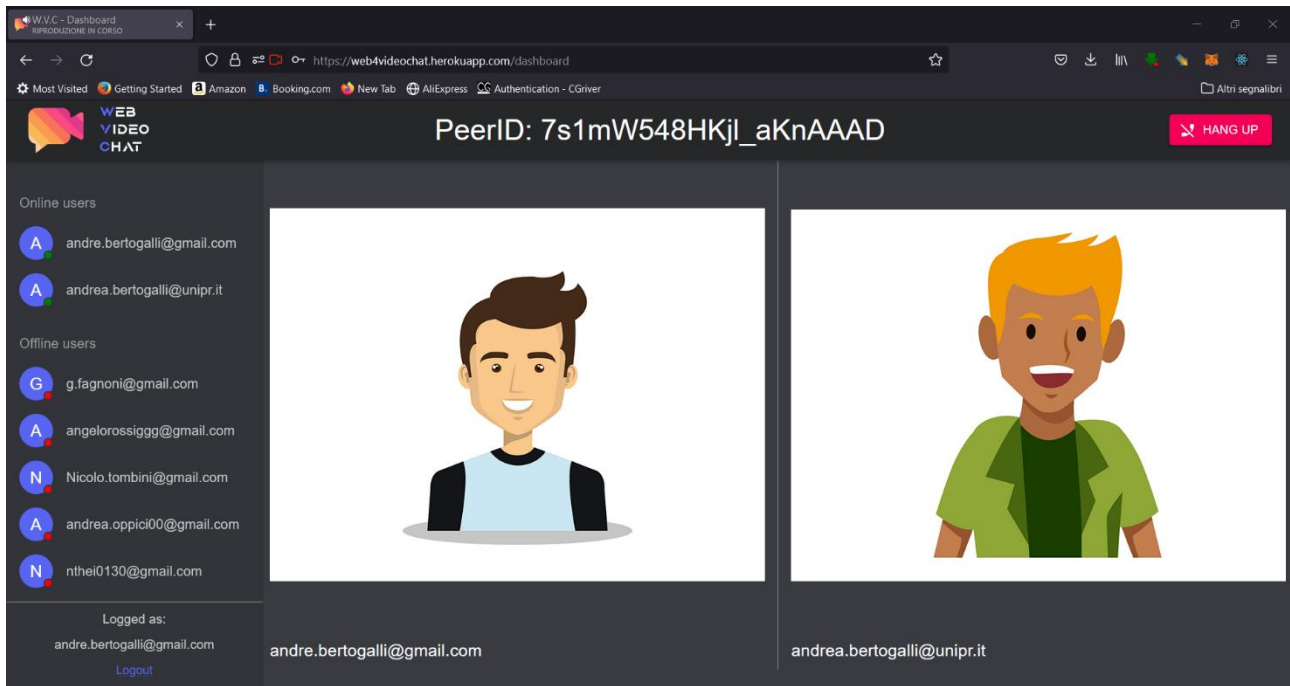


Se un utente è online e selezionato può essere effettuata una videochiamata premendo sul bottone "call" in alto a destra.

A questo punto, una volta premuto call, viene mandata una notifica all'utente selezionato, il quale può accettare la videochiamata oppure rifiutarla, mediante i bottoni answer e decline.



Una volta avviata la videochiamata, viene mostrata la ripresa dell'utente con cui si ha in corso la chiamata sulla destra con il suo username in basso. Premendo "hang up" in alto a destra si può terminare la chiamata.



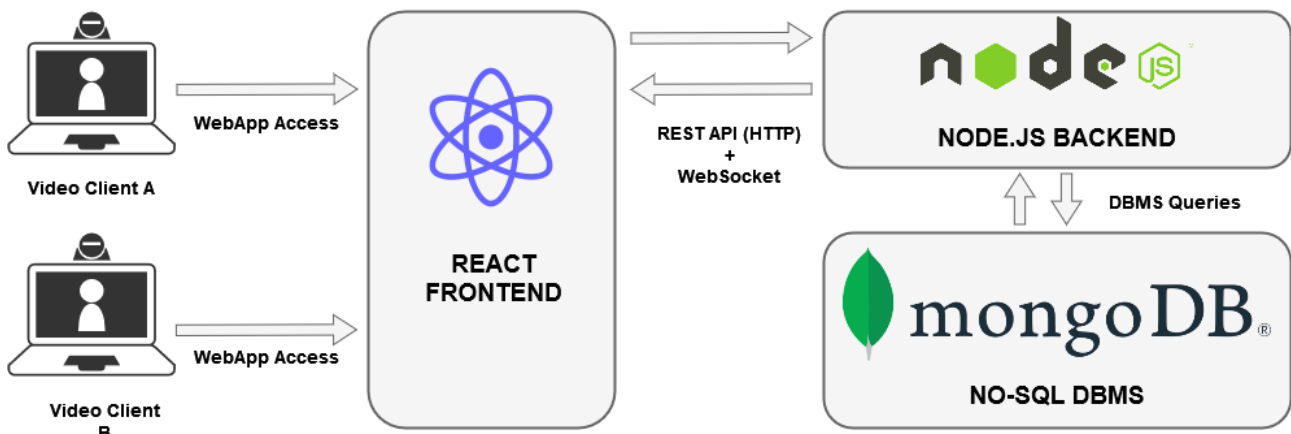
NOTA:

Se l'utente che ha effettuato l'accesso ha premuto "remember me", qualora ricarichi la pagina, chiuda il browser o semplicemente il tab rimarrà comunque loggato, per ritornare alla pagina di login deve premere il bottone di logout.

## SVILUPPO DEL PROGETTO

Il progetto è suddiviso in due parti principali, il **Backend** e il **Frontend**. La parte di **Backend** è fondamentalmente un server HTTP che mette a disposizione mediante API REST varie funzionalità al Frontend, come per esempio il login ecc... inoltre si interfaccia con il DBMS e funge anche da server per il protocollo WebSocket che è stato utilizzato per la parte di videochat mediante la libreria Socket.io. Il Frontend invece include solamente la GUI e permette all'utente di interagire con l'applicazione.

Di seguito viene riportato lo schema generale di funzionamento del progetto:



## TECNOLOGIE UTILIZZATE

### BACKEND

Il Backend ha tre funzionalità, quella di server HTTP, quella di gestione dei vari socket per l'interazione tra i vari utenti e infine quella di accesso alla base di dati (MongoDB) che contiene i vari utenti. Per lo sviluppo sono state utilizzate le seguenti tecnologie:

- **Express:** Express.js è un framework per applicazioni web per Node.js serve per creare web applications e API ed ormai definito il server framework standard de facto per Node.js.
- **Socket.io (server):** è una libreria che permette comunicazioni a bassa latenza, bidirezionali basate su eventi tra un client e un server, è costruita sul protocollo WebSocket.
- **Nodemon:** un tool molto utile durante lo sviluppo del server in quanto permette di mantenere aggiornato il server senza doverlo riavviare, funziona in maniera simile al rendering di react, ma per applicazioni node.
- **Mongodb (client):** Dato l'utilizzo di MongoDB come DBMS è stato necessario installare su node il modulo che ne permettesse l'interazione.

### FRONTEND

Il frontend è stato interamente sviluppato in **React**, con aggiunta di alcuni moduli:

- **Axios:** modulo estremamente utile che permette di effettuare in maniera estremamente efficace qualsiasi richiesta http e largamente utilizzato con React.
- **Material-ui:** questa libreria sviluppata da Google nel 2014 ed è attualmente la libreria appunto usata nei Frontend Google, mette a disposizione una vasta gamma di componenti grafici integrabili con React.
- **Socket.io (Client):** la parte client della libreria usata nel backend.
- **React-router-dom:** questo modulo è stato di estrema importanza per quanto riguarda la gestione degli accessi alle pagine e ai componenti, integrandosi con il metodo di autenticazione sviluppato.

NOTA: Sono stati usati altri moduli minori quali: react-audio-player per facilitare la riproduzione della suoneria delle chiamate, js-sha256 utile per generare hash sha256, vari set di icone ecc...

