

# Documentazione MuscleLab

Fanci Francesco Paolo, 2064829

Chieppa Lorenzo, 2062709

## 1 Struttura del progetto

Il progetto **MuscleLab** è organizzato secondo la seguente struttura:

```
BACKEND/
    server.js

PUBLIC/
    CSS/
    GRUPPI/
    IMG/
    JS/
    calc.html
    challegne.html
    diz.html
    gen.html
    home.html
    login.html
    map.html
    scheda-detail.html
    user.html
```

## 2 Descrizione della struttura

### BACKEND

Contiene il backend del progetto, sviluppato in Node.js. Al suo interno si trova il file `server.js`, responsabile della gestione delle richieste HTTP, tra cui:

- autenticazione e registrazione utenti;
- gestione delle sessioni;
- salvataggio e recupero dei dati relativi agli allenamenti;
- interfacciamento con eventuali database o moduli esterni.

### PUBLIC

Racchiude tutti gli elementi legati al frontend del sito. Al suo interno troviamo:

- **CSS/**  
Contiene i fogli di stile del sito. Sono presenti due file CSS principali:
  - uno per le pagine generali (es. `home.html`);
  - uno dedicato alle sottopagine, come la visualizzazione di schede salvate o gli esercizi di un gruppo muscolare.

- **GRUPPI/**

Contiene le pagine HTML alle quali l'utente viene reindirizzato cliccando su un gruppo muscolare all'interno del dizionario degli esercizi.

- **IMG/**

Include tutte le immagini, i loghi e i video utilizzati nelle varie sezioni del sito.

- **JS/**

Contiene gli script JavaScript associati a ciascuna pagina HTML. Ogni file gestisce le interazioni e le funzionalità dinamiche della relativa pagina.

- **File HTML**

I file `.html` rappresentano le varie pagine del sito, tra cui:

- `home.html`: homepage del sito;
- `login.html`, `user.html`: gestione degli utenti;
- `diz.html`, `map.html`: visualizzazione del dizionario degli esercizi;
- `challegne.html`, `gen.html`, `calc.html`: contenuti e funzionalità aggiuntive;
- `scheda-detail.html`: dettagli delle schede di allenamento.

## Nota Finale

All'interno della cartella **BACKEND** sono inoltre presenti alcuni file e directory fondamentali per il funzionamento dell'applicazione Node.js:

- `package.json` — Contiene le informazioni di configurazione del progetto, tra cui nome, versioni delle dipendenze e script di avvio.
- `package-lock.json` — Generato automaticamente da `npm`, garantisce che le stesse versioni delle dipendenze vengano installate in ambienti diversi.
- `node_modules/` — Directory creata da `npm install`, contiene tutte le librerie di terze parti necessarie all'esecuzione del progetto.