

Lab 10 – Autenticazione

L'obiettivo di questo laboratorio è di implementare un meccanismo di autenticazione per il social network sviluppato durante il corso, utilizzando il modulo **Flask-Login** (<https://flask-login.readthedocs.io/en/latest/>). In particolare, si richiede di:

1. Nel database, è necessario creare una tabella chiamata **UTENTI**, con i seguenti campi: ID (chiave primaria, impostata come auto-incremento), nome, cognome, username (con vincolo "UNIQUE"), e-mail (con vincolo "UNIQUE") e password.
2. Nella schermata principale dell'interfaccia utente, è necessario inserire un form per l'accesso. Questo modulo dovrebbe includere due campi: uno per lo username e uno per la password. In aggiunta, è richiesto un pulsante per inviare la richiesta di accesso (login). Contestualmente, creare una route dedicata sul server che si occupi di controllare i dati inseriti dall'utente ed eventualmente autenticarlo nel social network tramite **Flask-Login**.
3. Sotto il modulo di accesso, includere un link che, nel caso in cui un utente non sia ancora registrato, lo indirizzi verso una pagina dedicata alla registrazione. Questa pagina conterrà un modulo completo per l'inserimento di tutti i dati necessari per registrare un nuovo utente sulla piattaforma, con validazioni effettuate lato client.
Contestualmente, creare una route dedicata sul server che gestisca la raccolta dei dati inseriti nel modulo di registrazione e l'aggiunta del nuovo utente nel database. Si raccomanda di utilizzare **werkzeug.security** (<https://werkzeug.palletsprojects.com/en/3.0.x/>) per eseguire l'hashing della password prima di salvarla nel database.
4. Effettuare modifiche per limitare le azioni degli utenti non autenticati: impedire la creazione di nuovi post e commenti. Per gli utenti autenticati, la navbar mostrerà un messaggio di benvenuto e un link/bottone per il logout. Inoltre, i post dell'utente saranno evidenziati con uno sfondo colorato e sarà possibile aggiungere nuovi post e commenti.
Assicurarsi che queste modifiche siano applicate sia sull'interfaccia utente, dove i componenti per creare post e commenti saranno nascosti agli utenti non autenticati, sia sul server. Sul server, sarà necessario aggiungere i decorator `@login_required` alle funzioni coinvolte per evitare l'accesso da parte di utenti non autenticati.
5. Alla pressione del bottone per il logout, effettuare il logout dell'utente dalla piattaforma, servendosi di una opportuna route sul server.