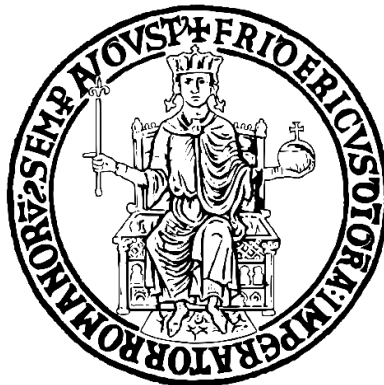


UNIVERSITÀ DEGLI STUDI DI NAPOLI “FEDERICO II”
DIPARTIMENTO DI INGEGNERIA ELETTRICA E TECNOLOGIE
DELL’INFORMAZIONE



CORSO DI LAUREA IN INFORMATICA
INSEGNAMENTO DI LABORATORIO DI PROGRAMMAZIONE
ANNO ACCADEMICO 2020/2021

Sviluppo di un social network minimalista

Autore:

Crescenzo Lucio CICATIELLO
N86003457
cr.cicatiello@studenti.unina.it

Docenti:

Prof. Bastien MAUBERT
Prof. Aniello MURANO

Sommario

Progetto.....	2
Approccio alla soluzione.....	2
Traccia in breve	2
Soluzione	2
Funzionalità e descrizione	3
Strutture dati utilizzate.....	4
Struct data	4
Struct utente.....	4
Struct seguiti.....	4
Esempio di esecuzione	4

Progetto

Approccio alla soluzione

Traccia in breve

L'obiettivo del nostro progetto è la realizzazione di un social network minimalista con varie e semplici funzionalità iniziali tra cui:

- **Sign up** L'utente può creare un nuovo account.
- **Log in** : L'utente sceglie di fare il log in.
- **Delete account** : L'utente sceglie di eliminare il suo account.

Al seguito dell'avvenuto login l'utente ha accesso alle seguenti funzionalità:

- Visualizzare il nome di tutti gli utenti.
- Cercare se esiste un utente.
- Decidere di seguire un utente.
- Fare un post
- Vedere tutti i post di un utente.
- Vedere l'ultimo post di ogni utente seguito.
- Fare il log out.

Le uniche limitazioni applicare all'intero social sono:

- Post di solo testo.
- Massimo numero di post per account 10.

Soluzione

Conservazione dei dati

L'approccio alla soluzione è stato diviso in: conservazione dei dati dell'utente, dei post e dei seguiti di un'utenza.

- **Dati utente**: i dati dell'utente sono stati unificati in un'unica struct di nome "data" al cui all'interno ho inserito username, password, post e seguiti. Quest'ultima è usata in una struct "utente" dove si conoscono i dati e la posizione del prossimo utente.
- **Post**: i post sono stati salvati in un array bidimensionale con massimo dieci righe ed una lunghezza massima del post di 400 caratteri.
- **Seguiti**: I seguiti sono stati salvati in una coda dove ogni seguito conosce l'utente (si punta direttamente alla lista di utente per evitare la ridondanza delle informazioni) quindi i suoi dati ed il prossimo seguito.

Funzionalità in breve

Ho separato le funzionalità del social in due menù separati prima e dopo il login dell'utente. Prima del login (Figura 1) l'utente ha a disposizione le seguenti funzionalità:



Figura 1 – Menu prima del log in

Mentre dopo aver effettuato il log in con username esistente e password si entra in un secondo menù (Figura 2) dove abbiamo a disposizione le seguenti funzionalità richieste dal social:

- Visualizzazione del nome degli utenti.
- Ricerca di un utente.
- Aggiunta di un utente ai seguiti.
- Creazione di un nuovo post.
- Mostra gli ultimi dieci post di un utente.
- Mostra l'ultimo post dei seguiti.
- Mostra l'ultimo post di ogni utente.
- Log out torna al menu precedente.

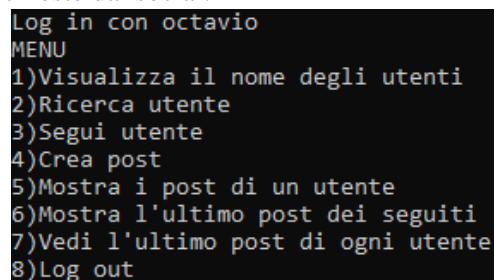


Figura 2 - Menu dopo il log in

Funzionalità e descrizione

Funzioni	Descrizione
Menu_login	Viene visualizzato un menu dove l'utente ha possibilità di scelta sulle azioni da fare dopo il login.
Sign up	Viene creata una nuova utenza dove viene richiesto username (che viene verificato se già esistente), la password e la conferma password.
Aggiungi utente	Prende in input un'utenza che aggiunge alla lista già esistente.
Stampa utenti	Stampa il nome di tutti gli utenti scorrendo la lista.
Controllo utenza	Verifica l'esistenza di un'utenza in lista e la restituisce.
Controllo password	Verifica che la password inserita corrisponda con quella associata all'utenza.
Log in	Verifica che l'utenza esista e che la password sia corretta in caso affermativo
Ricerca utente	Verifica se l'utenza sia corretta.
Crea post	Creo un nuovo post per l'utente che ha effettuato il log in.
Stampa ultimo post utenti	Stampo l'ultimo post di tutti gli utenti
Mostra post	Mostra gli ultimi dieci post dell'utente inserito.
Segui utente	Inserisco un nuovo utente ai seguiti dell'utente che ha effettuato il log in.
Vedi post seguiti	Mostra l'ultimo post di tutti gli utenti seguiti.
Init	Inizializza il social con cinque utenti con dieci post ciascuno.

Elimina utente	Elimina l'utenza ed i dati associati.
Elimina seguito	Elimina l'utenza che si sta per eliminare dai seguiti degli altri utenti.
Controllo seguiti	Verifica che un'utenza non sia già seguita.

Strutture dati utilizzate

Le strutture utilizzate per la conservazione delle informazioni dell'utente come precedentemente detto sono state divise in tre parti differenti che elencherò in forma tabellare con le rispettive descrizioni. Come lunghezze massime per i vari campi ho usato **16 come lunghezza fissa di password e username e 400 come lunghezza massima di un post.**

Struct data

Lo struct data è stato implementato per contenere le informazioni dell'utente come vediamo da tabella.

Nome	Definizione
char username[USER_LENGTH]	Contiene l'username dell'utente max 16 char.
char password[USER_LENGTH]	contiene la password dell'utente max 16 char.
char post[10][POST_LENGTH]	array bidimensionale contenente massimo 10 post con una lunghezza di 400 char.
seguiti * seguiti	contiene il puntatore alla testa della lista dei seguiti.

Struct utente

Lo struct utente è stato implementato per poter creare una lista di puntata a variabili di tipo struct utente.

Nome	Definizione
data infoutente	Contiene username, password, post e seguiti di un utente.
utente * next	Contiene l'indirizzo del prossimo utente della lista.

Struct seguiti

Lo struct seguiti è stato implementato per poter creare una lista di puntatori a variabili di tipo struct seguiti. Inoltre, ho evitato di aggiungere il nome per evitare la ridondanza dei dati ed ho direttamente puntato all'utenza seguita.

Nome	Definizione
utente * seguito	Contiene l'indirizzo dell'utente seguito di tipo utente.
seguiti * next	Contiene l'indirizzo del prossimo utente della lista dei seguiti.

Esempio di esecuzione

