**Struttura del sito web:**

Il sito ammette due tipologie di utenti, che si suddividono in utenti visitatori e utenti registrati.

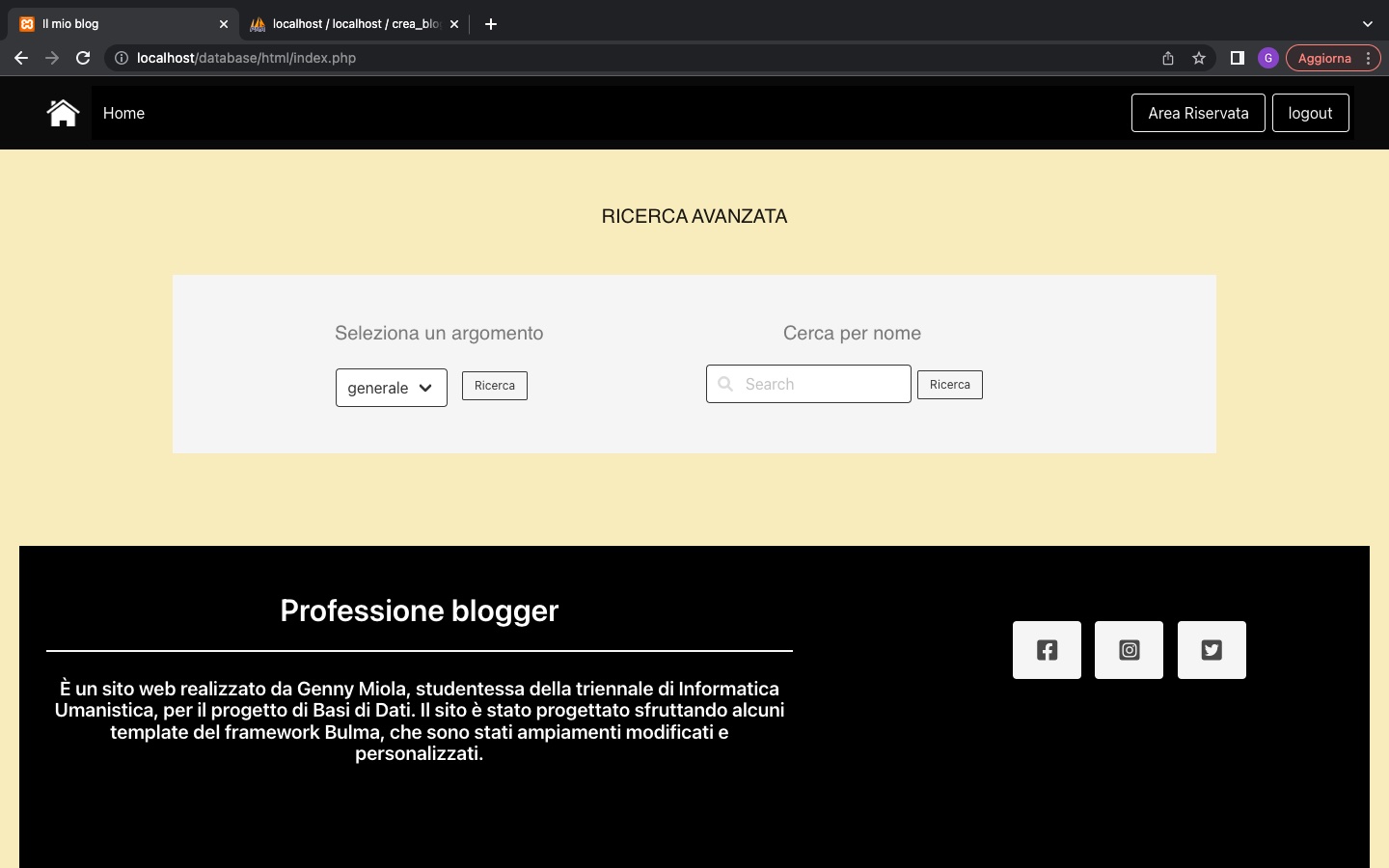
I primi sono coloro che non hanno compilato il form di registrazione alla piattaforma e la cui fruizione del sito si limita alla lettura e visione dei contenuti, dato che non hanno la possibilità di interazione attraverso like, commenti o creazione di contenuti.

I secondi sono quelli che, invece, si sono registrati alla piattaforma e che dispongono l’accesso ad un’area riservata, nella quale possono visionare e modificare i propri dati personali, creare, modificare e eliminare blog e post e nominare dei coautori per i propri blog, con l’opzione di poter revocare questa carica successivamente. Un utente registrato può, inoltre, inserire commenti all’interno dei propri e dei post altrui, mettere like e, in base al numero di interazioni effettuate fra inserimento di commenti e creazione di post, superare vari livelli e sbloccare nuove grafiche e ottenere una funzionalità aggiuntiva in fase di creazione dei blog e dei post: la possibilità di inserire un nuovo argomento/sottoargomento personalizzato.

**HOME PAGE**

Nella home page vengono mostrati gli ultimi 8 post che sono stati pubblicati, dal più recente al più antico. L’utente (visitatore/registrato) può cliccare sui post per leggerli oppure sfruttare la barra di navigazione per cercare all’interno del sito i blog per argomento oppure usare la ricerca per nome, cercando il titolo di un post o di un blog o parte di esso e accedervi.





**POST**

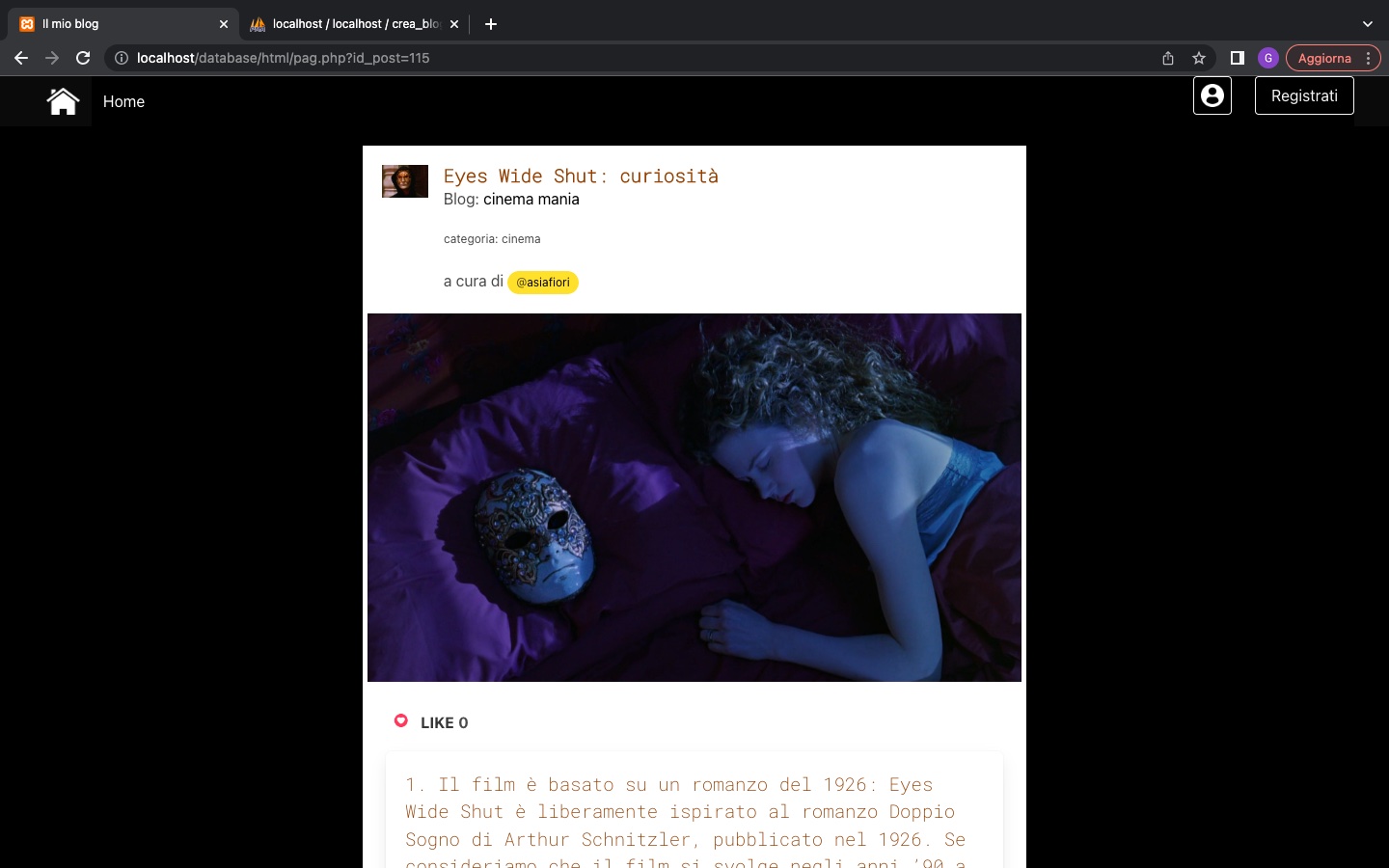
L’aspetto della pagina dedicata al post riprende quello di un post di Instagram.

Dalla pagina del post è possibile vedere in alto il nome del blog di appartenenza e, cliccandoci, accedere alla pagina del blog e lo username dell’utente che ha creato il post, anche questa scritta è cliccabile e conduce alla pagina profilo dell’utente.

Durante la fase di creazione di un post, l’utente ha la possibilità di caricare fino a due immagini: una di copertina, che verrà visualizzata nella pagina del post in alto a sinistra, e una come allegato, che sarà l’immagine centrale presente nella pagina. A ciascun post, l’utente assegna un titolo, che deve avere una lunghezza minima di 3 caratteri e un testo, che deve essere compreso fra i 50 e 2000 caratteri. L’utente può selezionare, inoltre, uno dei sottoargomenti disponibili per il macroargomento assegnato al blog di appartenenza del post, oppure, se ha superato tutti i livelli, può inserirne uno nuovo, che verrà di conseguenza aggiunto anche al database.

Per l’utente registrato la pagina del post presenta anche altre funzionalità, fra cui il pulsante per inserire un like al post con la possibilità di rimuoverlo premendo nuovamente il bottone like; il pulsante per inserire un commento al post, che potrà essere modificato successivamente e anche eliminato (per ciascun commento comparirà anche lo username dell’utente che lo ha pubblicato e la data e l’ora di pubblicazione); e il pulsante per la modifica del post.

L’utente visitatore potrà, invece, soltanto vedere il numero di like del post e i commenti pubblicati.

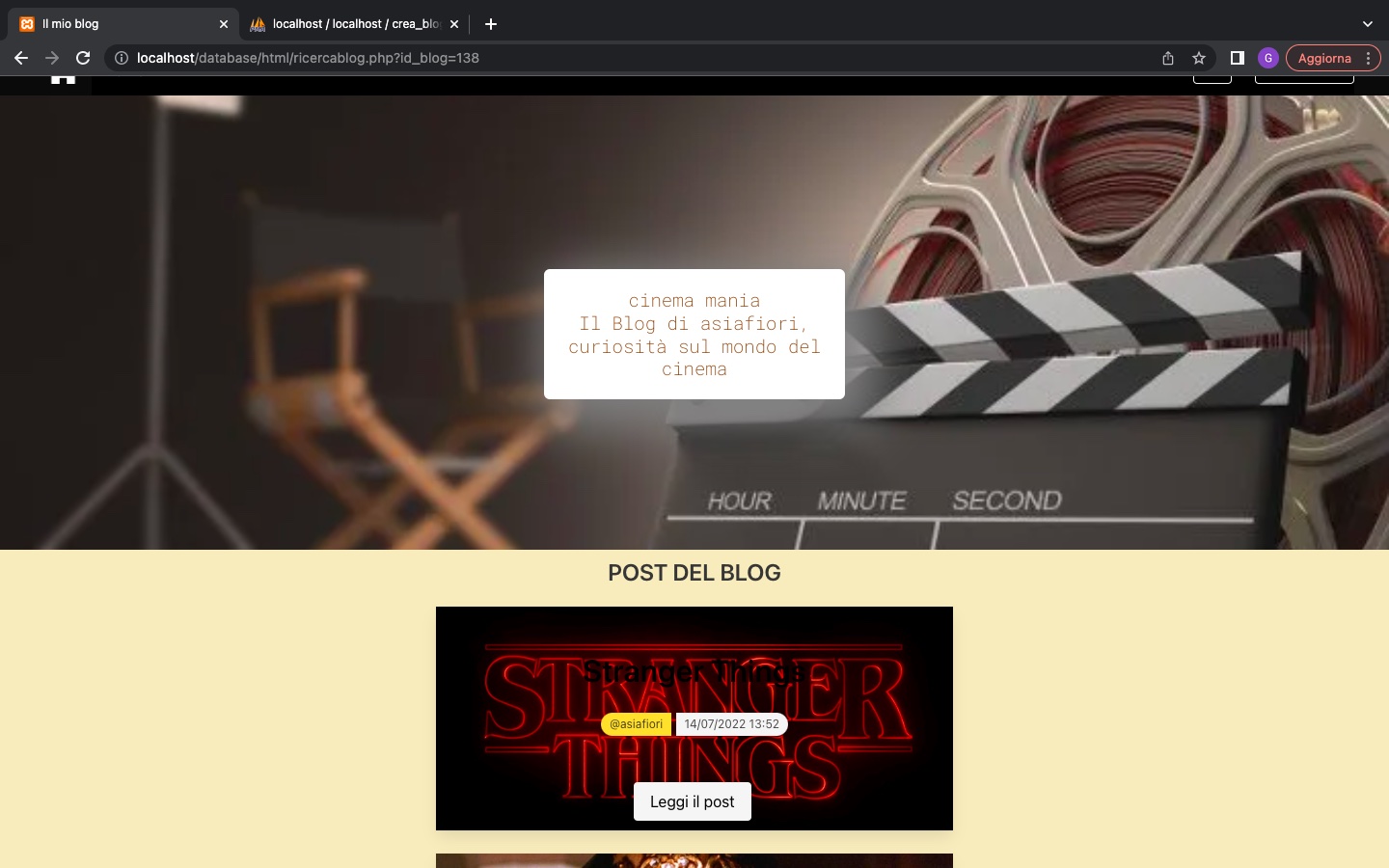


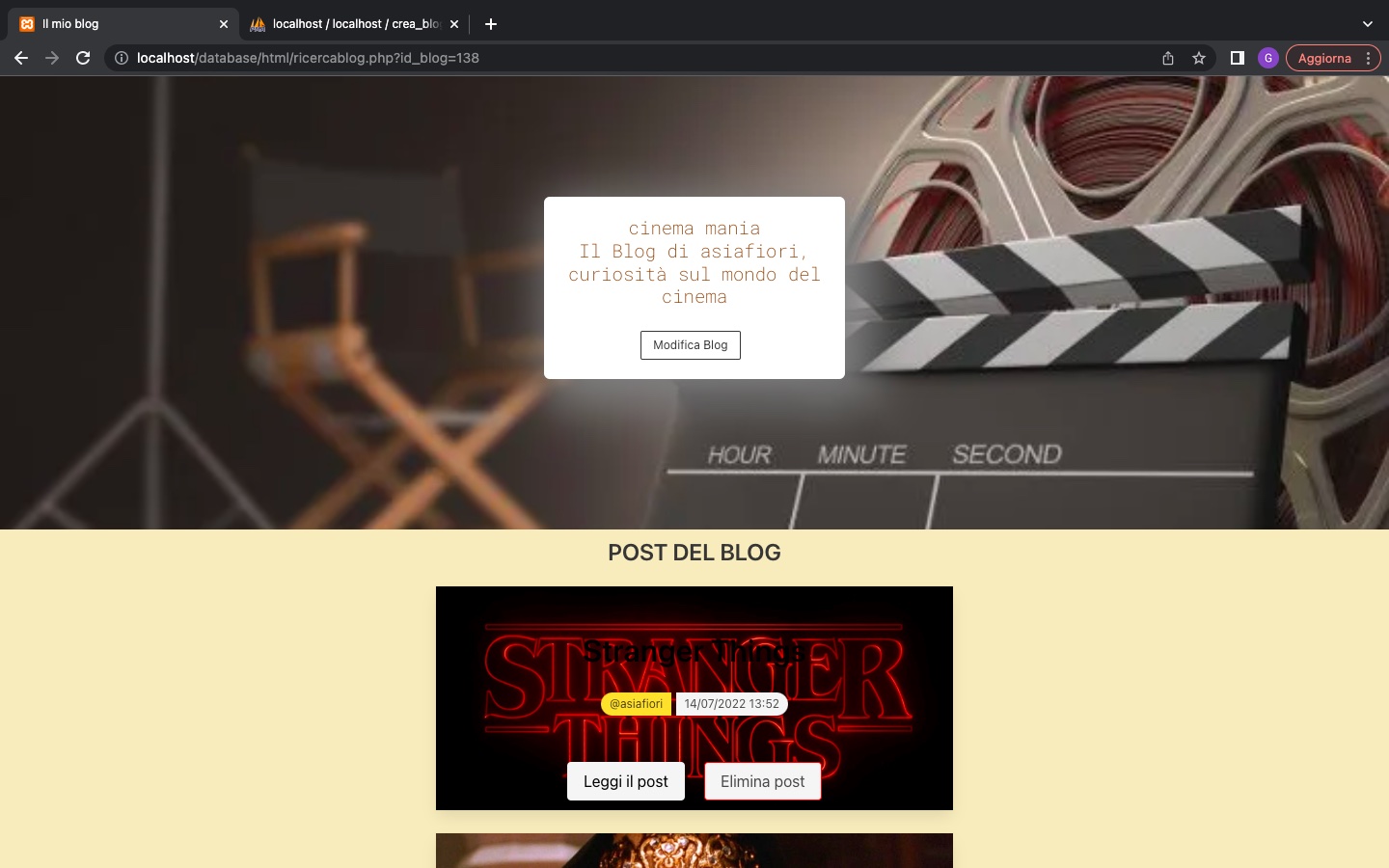
**BLOG**

L’utente registrato, in fase di creazione del blog, deve inserire un titolo, che non deve essere inferiore a 3 caratteri, una breve descrizione, anch’essa di lunghezza limitata e non inferiore a 20 caratteri, scegliere un’immagine di copertina (opzione facoltativa) e selezionare un argomento di appartenenza del blog, può anche scegliere uno stile per il proprio blog (la mancata selezione di uno stile in particolare farà sì che venga impostato uno stile predefinito) e nominare un coautore che avrà la possibilità di contribuire allo sviluppo e modifica del blog stesso, ma non potrà eliminarlo.

La pagina dedicata al blog presenta una sezione in alto in cui è presente l’immagine di copertina selezionata dall’utente per il proprio blog (se presente), il titolo, la descrizione del blog e lo username dell’utente che lo ha creato e, in basso, la lista dei post scritti per quel blog a cui si può accedere cliccando sul pulsante leggi.

Per l’utente registrato e proprietario del blog, nella pagina del blog compare anche il pulsante per la modifica del blog, attraverso il quale vi è la possibilità di cambiarne le caratteristiche e il pulsante per eliminare i vari post di cui è autore.

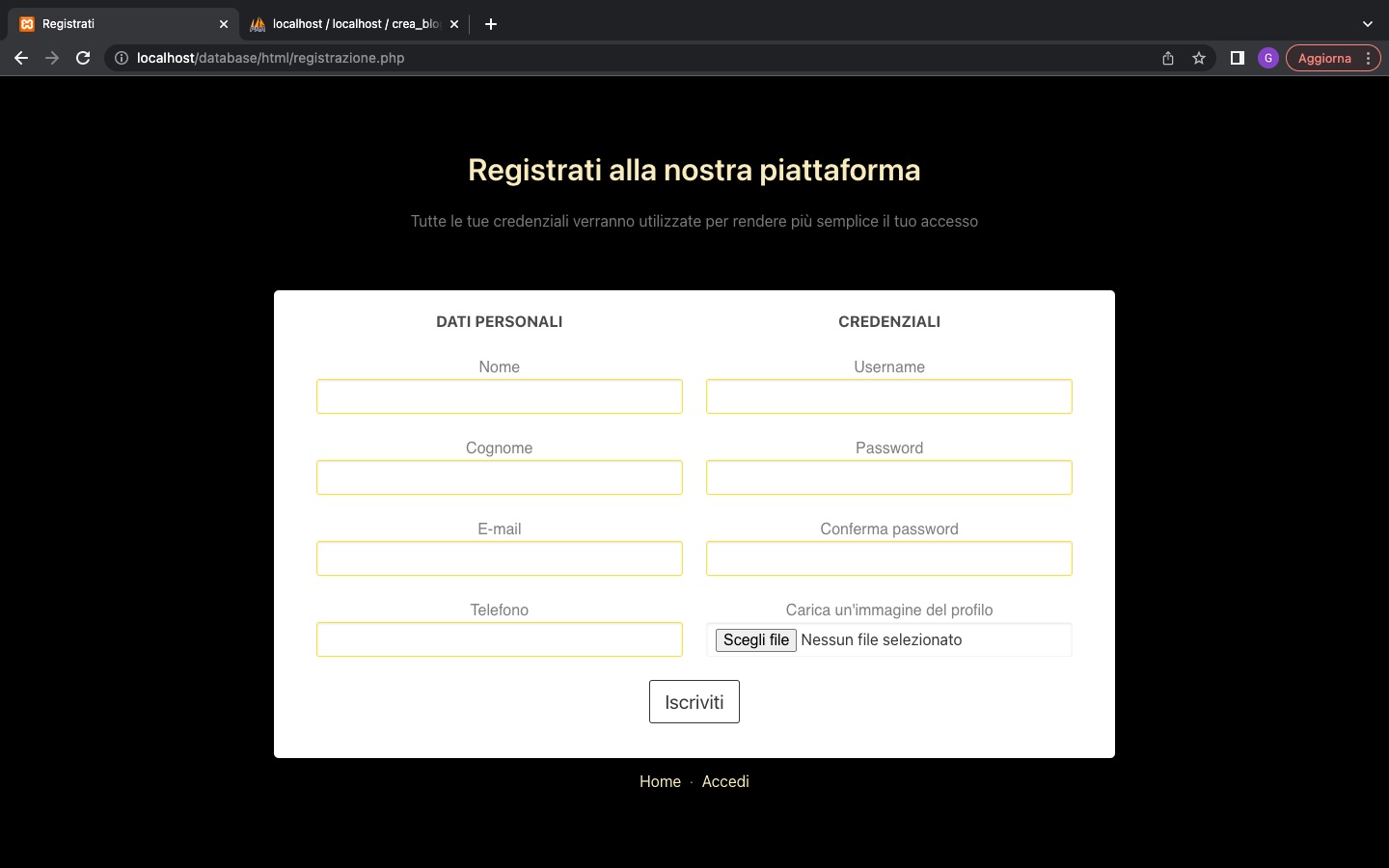




**REGISTRAZIONE**

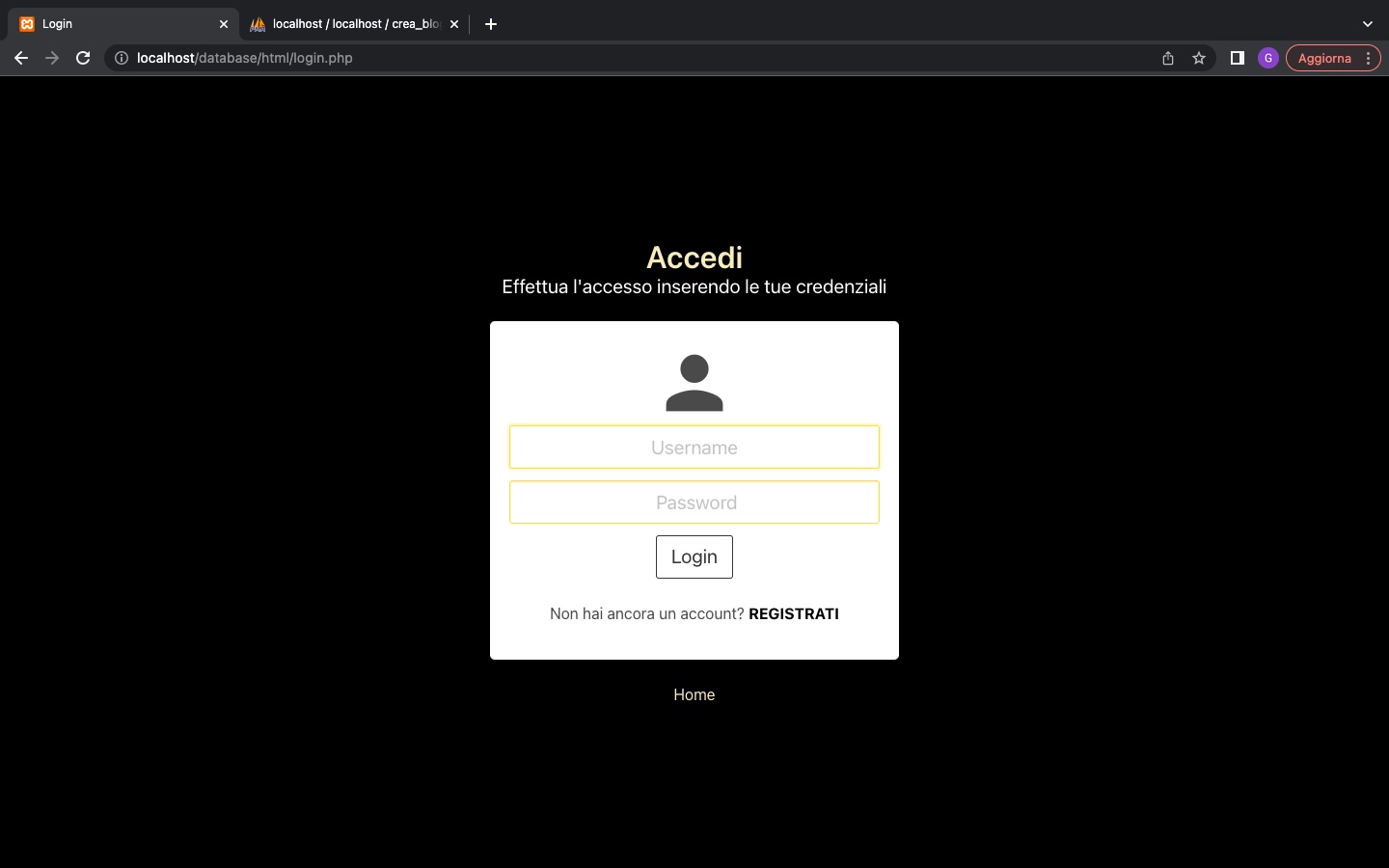
Per registrarsi alla piattaforma un utente deve inserire nome e cognome, un’email che rispetti un formato valido (deve presentare una @ all’interno, un punto e dopo il punto un numero di caratteri limitato da due a sei per il dominio di appartenenza), deve poi inserire un username (almeno di 6 caratteri), una password complessa che, per motivi di sicurezza, deve contenere almeno una maiuscola, un numero e un carattere speciale, e confermare la password inserita reinserendola nuovamente nel campo password. Per questi campi non sono ammessi valori vuoti.

I campi facoltativi che, invece, l’utente può decidere di inserire oppure no, sono il telefono (fisso o mobile, che rispetti anche in questo caso un formato valido e non presenti spazi e punti all’interno) e la selezione di un’immagine del profilo che verrà visualizzata nella pagina del profilo dell’utente. Una volta effettuata la registrazione, l’utente verrà reindirizzato alla home page del sito e potrà effettuare il login.



**LOGIN**

In fase di login, l’utente, che si è precedentemente registrato, deve inserire username e password corretti per accedere al sito. I campi sono obbligatori e non possono essere vuoti. Il valore inserito nei campi deve coincidere con quello inserito dall’utente nei corrispettivi campi durante la fase di registrazione.



**AREA RISERVATA**

L’utente registrato e loggato ha a disposizione un’area riservata. Nell’area riservata l’utente può modificare i propri dati personali: cambiare l’username, email, telefono e/o modificare la password, reinserendo quella attuale, con la quale ha effettuato l’accesso al sito, e inserendone una nuova che dovrà essere poi confermata nel campo conferma password. L’utente può, inoltre, creare nuovi blog, attraverso il pulsante “crea nuovo blog” e visionare i blog creati o per i quali è stato nominato coautore da altri utenti nella sezione “i miei blog”. Qui, si ha poi la possibilità di accedere alla pagina del blog, cliccando sul titolo, di modificare il blog, di creare un nuovo post per il blog oppure di eliminarlo, sfruttando gli appositi pulsanti.

L’area riservata presenta anche la sezione “i miei coautori” nella quale l’utente potrà visualizzare gli utenti che ha nominato come coautori e i blog per i quali lo ha fatto e avrà la possibilità di revocare il ruolo assegnatogli eliminandoli come coautori attraverso il pulsante “elimina”.

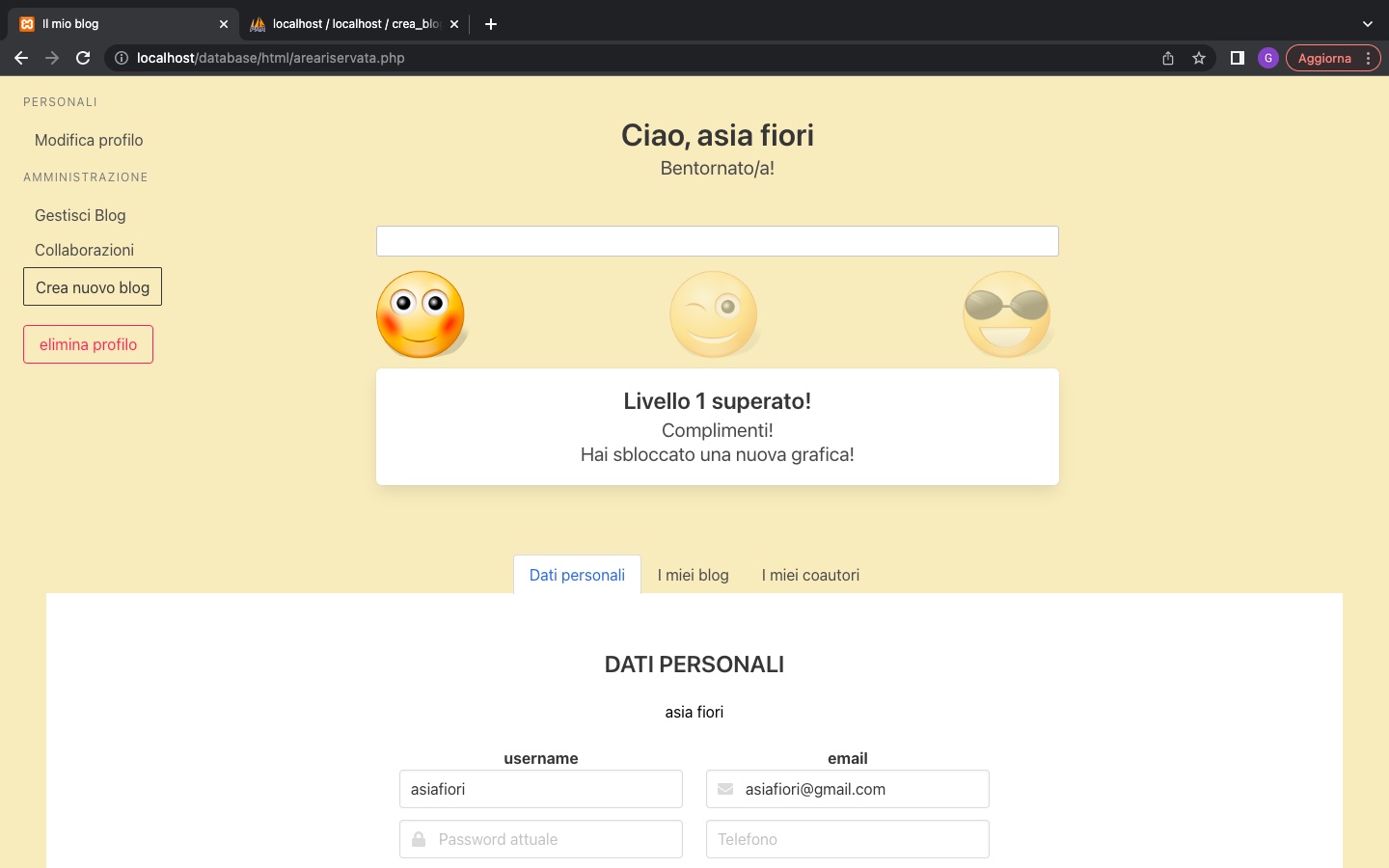
Infine, dall’area riservata l’utente può vedere i livelli che sono stati sbloccati, in base al suo grado di interattività. Vi sono, infatti, tre livelli rappresentati graficamente da 3 emoticon, che si colorano a mano a mano che l’utente avanza di livello, mentre la percentuale di avanzamento viene rappresentata da una barra che mostra “la distanza” per il raggiungimento del livello successivo e che, una volta completato ciascun livello, si svuota e riparte da zero.

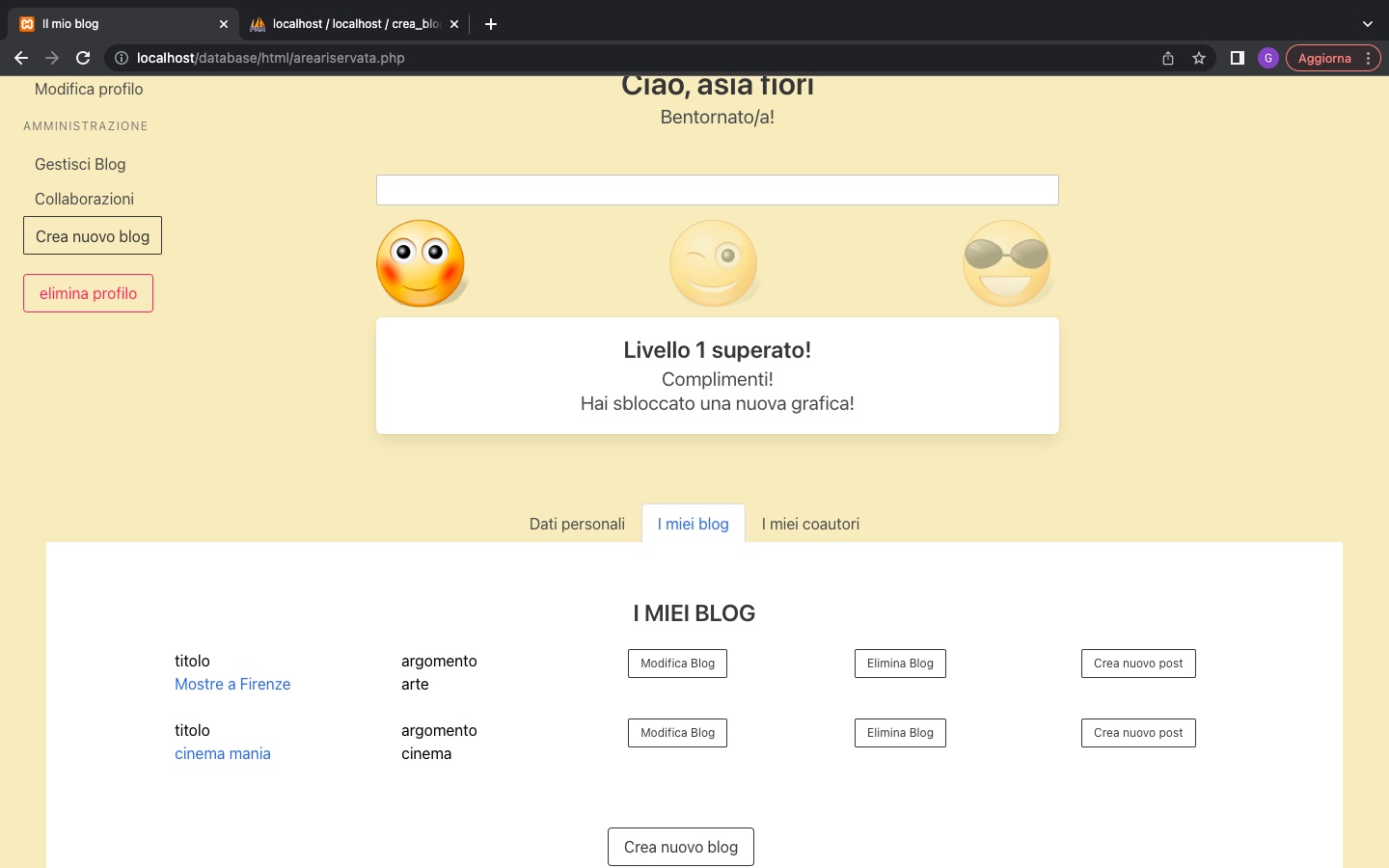
Ogni volta che l’utente sblocca un nuovo livello compare anche un box di congratulazioni, in cui vengono descritti i vantaggi acquisiti.

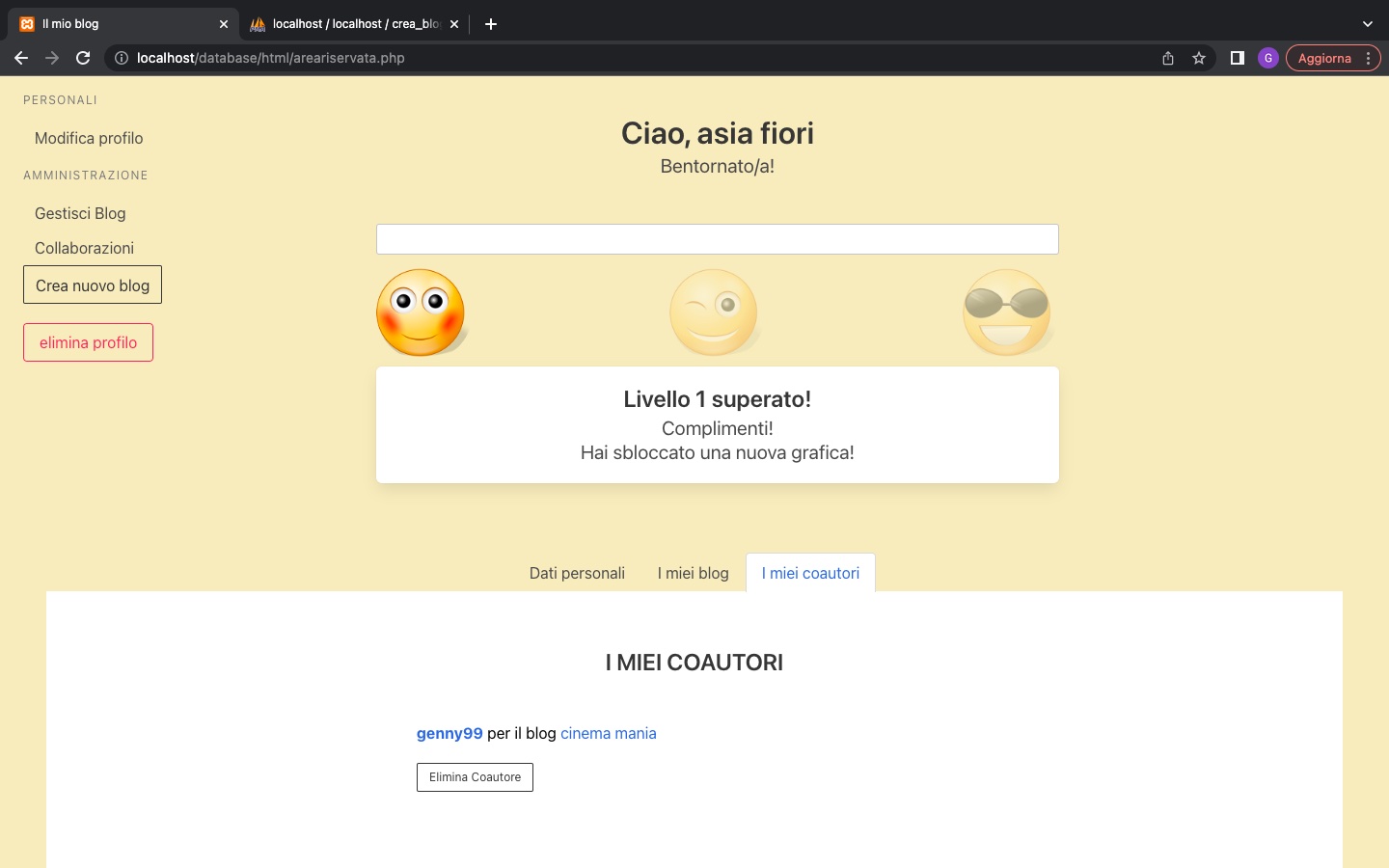
Il raggiungimento dei vari livelli si basa sul calcolo del numero di post e commenti inseriti dall’utente: la soglia è di 5 post e/o commenti per il primo livello, 8 per il secondo e 10 per il terzo.

Sbloccando il primo livello, l’utente ha a disposizione una nuova grafica visualizzabile durante la fase di creazione del blog, sbloccando anche il secondo livello, oltre ad una nuova grafica, l’utente può selezionare come argomenti e sottoargomenti per i propri blog e post quelli inseriti da utenti che abbiano già sbloccato il terzo livello, mentre, sbloccando il terzo livello, oltre ad avere un’ulteriore grafica, si aggiunge la possibilità di inserire nuovi argomenti e/o sottoargomenti sempre in fase di creazione del blog e/o del post che si andranno ad aggiungere a quelli già presenti nel database.

L’area riservata presenta, infine, il pulsante che permette all’utente di cancellare il proprio profilo: nel momento in cui l’utente preme questo pulsante, comparirà un messaggio di avviso attraverso cui l’utente potrà confermare la propria azione oppure annullarla.



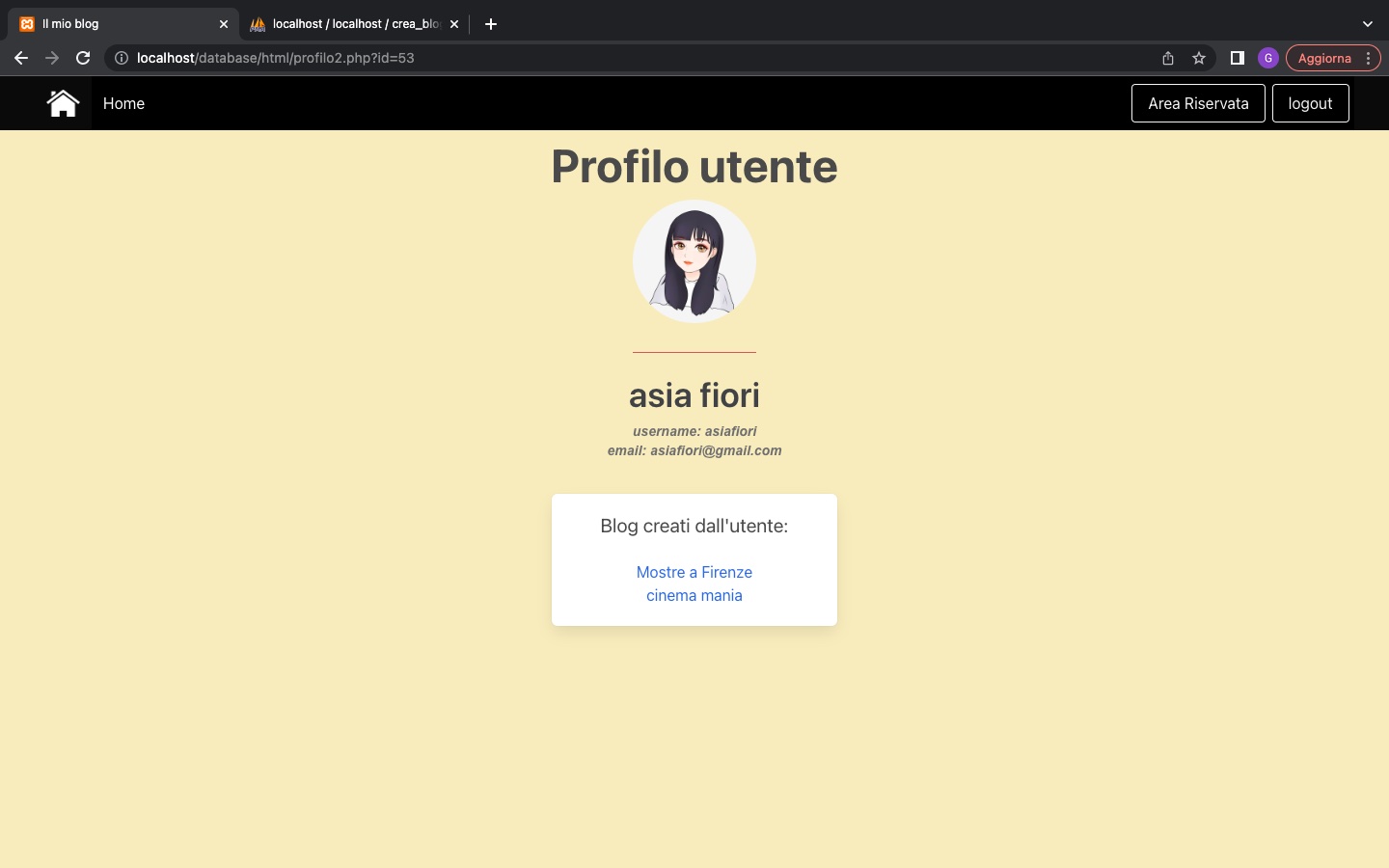




**PAGINA DEL PROFILO**

Attraverso la pagina di visualizzazione di un post, cliccando sullo username dell’utente che lo ha creato, è possibile, sia per l’utente registrato che per il visitatore, accedere alla pagina del profilo di quell’utente. La pagina del profilo presenta in alto un’immagine che sarà l’immagine del profilo inserita dall’utente in fase di registrazione, oppure, in assenza di essa, un’immagine di default.

Sotto vengono riportati alcuni dati dell’utente, come lo username e l’email e, infine, in basso viene riportata la lista dei blog creati da quell’utente: cliccando sul titolo di ciascuno di essi si accede alla pagina di visualizzazione del blog e dei relativi post.



**NAVBAR**

La barra di navigazione presenta a sinistra due pulsanti che riportano alla home pagine: cliccando sia sull’icona, sia su home, si verrà reindirizzati alla pagina index del sito.

A destra della navbar, invece, se l’utente è un utente che non si è registrato o loggato al sito vedrà i pulsanti per il login e la registrazione, altrimenti, se l’utente è già registrato e si è loggato al sito, questi pulsanti verrano sostituiti da i bottoni “area riservata” e “Logout”. Cliccando su “logout” l’utente verrà sloggato dal sito e, per accedervi, dovrà effettuare nuovamente l’accesso tramite la pagina di login.

F**UNZIONALITÀ DEL SITO:**

**REGISTRAZIONE:**

Un utente, che non abbia già effettuato la login, può accedere al form per registrarsi alla piattaforma attraverso l’apposito pulsante presente nella navbar.

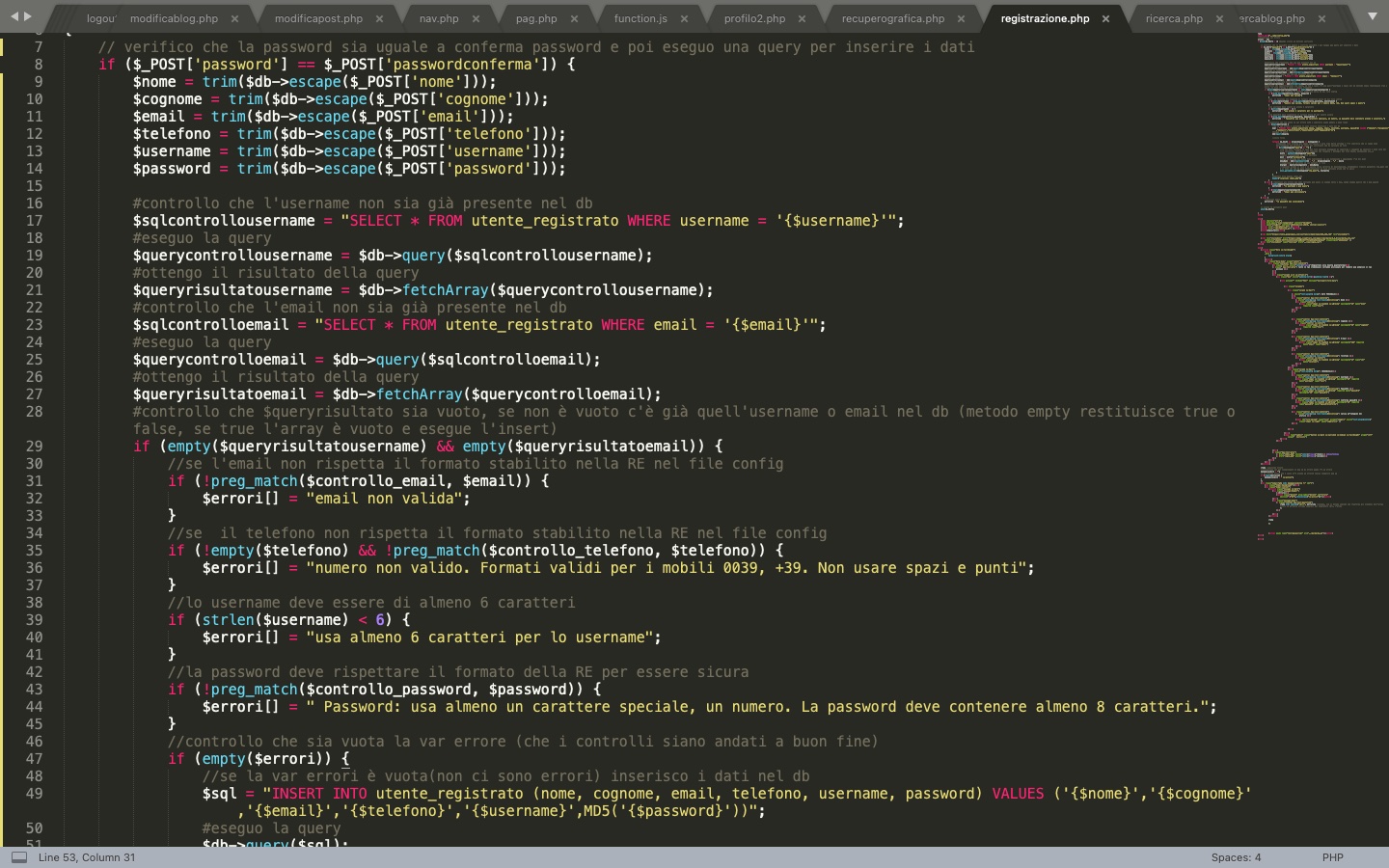
Le informazioni che devono essere inserite durante la fase di registrazione e che sono obbligatorie sono: nome e cognome, campi che hanno una lunghezza massima di 20 caratteri; email, campo con dimensione massima di 100 caratteri, che deve rispettare il formato descritto dall’espressione regolare "/[A-z0-9\.\+\_-][+@[A-z0-9\.\_-]+\.[A-z]{2,6}$/](mailto:+@[A-z0-9\._-%5d+\.%5bA-z%5d%7b2,6%7d$/)" e che non deve essere già presente nel database, qualora la mail sia già stata utilizzata comparirà all’utente un messaggio che lo avverte di usare una mail diversa; username, campo che deve avere una lunghezza minima di 6 caratteri e massima di 20, e di cui, come per l’email, si controlla che non sia già presente nel database e, in caso contrario, si avverte l’utente di modificare lo username, perché quello scelto è già utilizzato; password, campo con lunghezza massima di 20 caratteri, di cui si controlla che sia lunga almeno 8 caratteri, che contenga almeno un carattere speciale, una maiuscola e un numero, usando la funzione preg\_match con l’espressione regolare "/^(?=.\*\d)(?=.\*[#$@!%&\*?.\_])[A-Za-z\d#$@!%&\*?.\_]{8,}$/"; conferma password, campo lungo massimo 20 caratteri, in cui si controlla che ciò che è stato inserito nel campo password coincida con quanto inserito nel campo conferma password e, in caso non coincidano, si avverte l’utente con un messaggio di errore.

I campi facoltativi sono, invece, telefono, che ha una lunghezza massima di 10 caratteri e di cui si controlla che rispetti il formato stabilito dalla RE "/^((00|\+)39??)??(05|3)\d{2}\d{6,7}$/", per cui un numero di telefono non ammette spazi o punti, ma ammette in apertura per i telefoni mobili lo 0039 e il +39; e la selezione di un file per caricare un’immagine del profilo dell’utente che verrà visualizzata nella pagina profilo2.php.

Per tutti i campi utilizzo la funzione trim, per eliminare gli spazi bianchi e far sì che i campi non restino vuoti, la funzione escape, che utilizza il metodo mysqli\_real\_escape\_string che sostituisce apice, doppio apice, backslash e la stringa nulla con le rispettive sequenze di escape in modo tale da evitare eventuali attacchi e nella quale ho definito anche la funzione htmlspecialchars() , così da convertire i caratteri speciali di una stringa e trasformarli nei corrispondenti codici HTML.

Una volta effettuata la registrazione, se questa va a buon fine, l’utente verrà reindirizzato nella pagina index per fare la login.

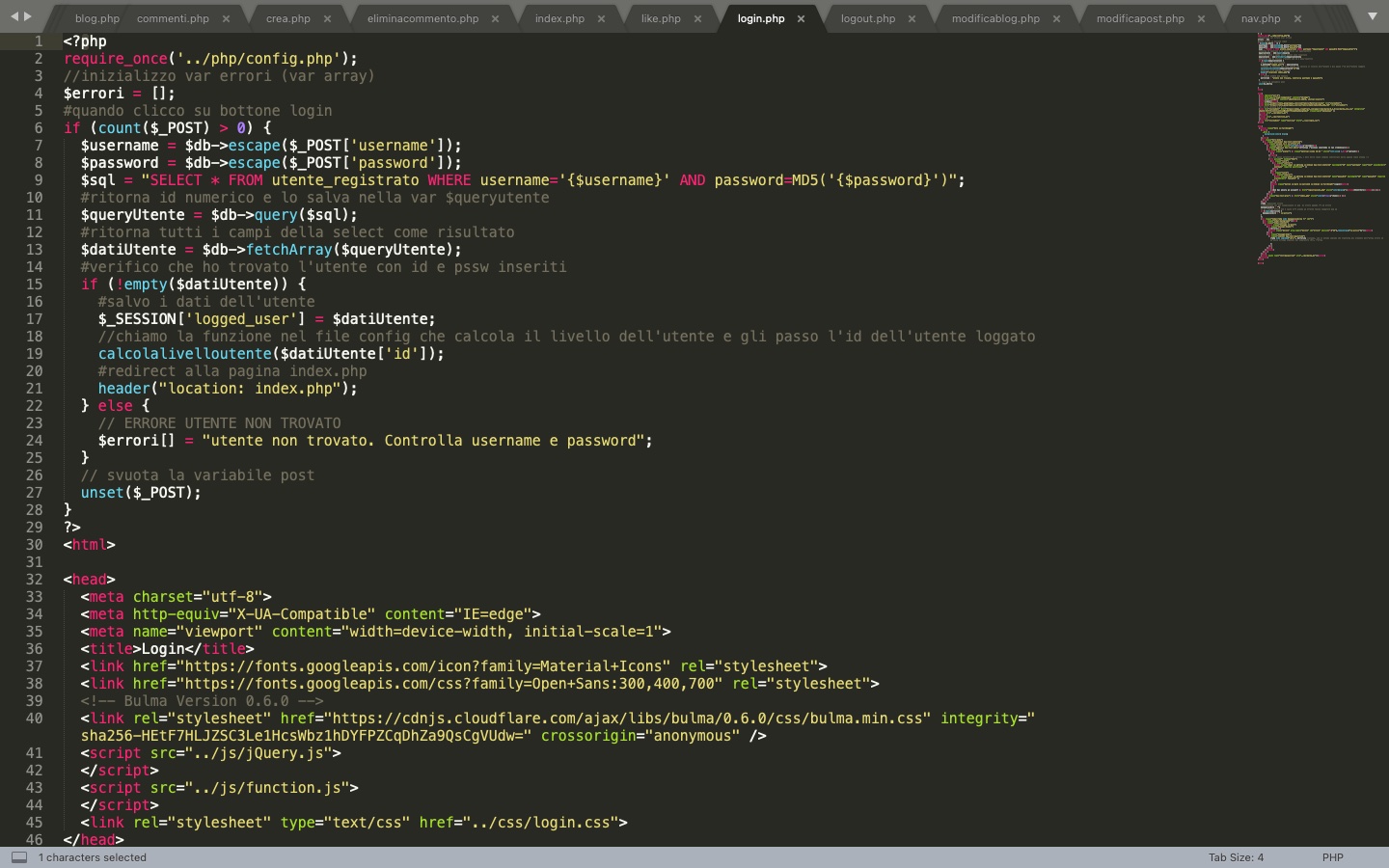
Un esempio dei controlli che vengono effettuati in fase di registrazione:



**LOGIN**

I campi che l’utente deve compilare per effettuare la login al sito sono: username e password. I due campi sono obbligatori, se uno dei due o entrambi resta vuoto, comparirà all’utente un messaggio di avviso. Controllo poi che l’username e la password siano esatti per quell’utente e, in tal caso, l’utente verrà reindirizzato alla index come utente loggato e i suoi dati verranno salvati in sessione. Durante il login, uso anche la funzione calcolalivelloutente() per calcolare il livello a cui l’utente si trova e che sarà visualizzato nell’area riservata.

Anche in questo caso, uso la funzione escape per i campi username e password.



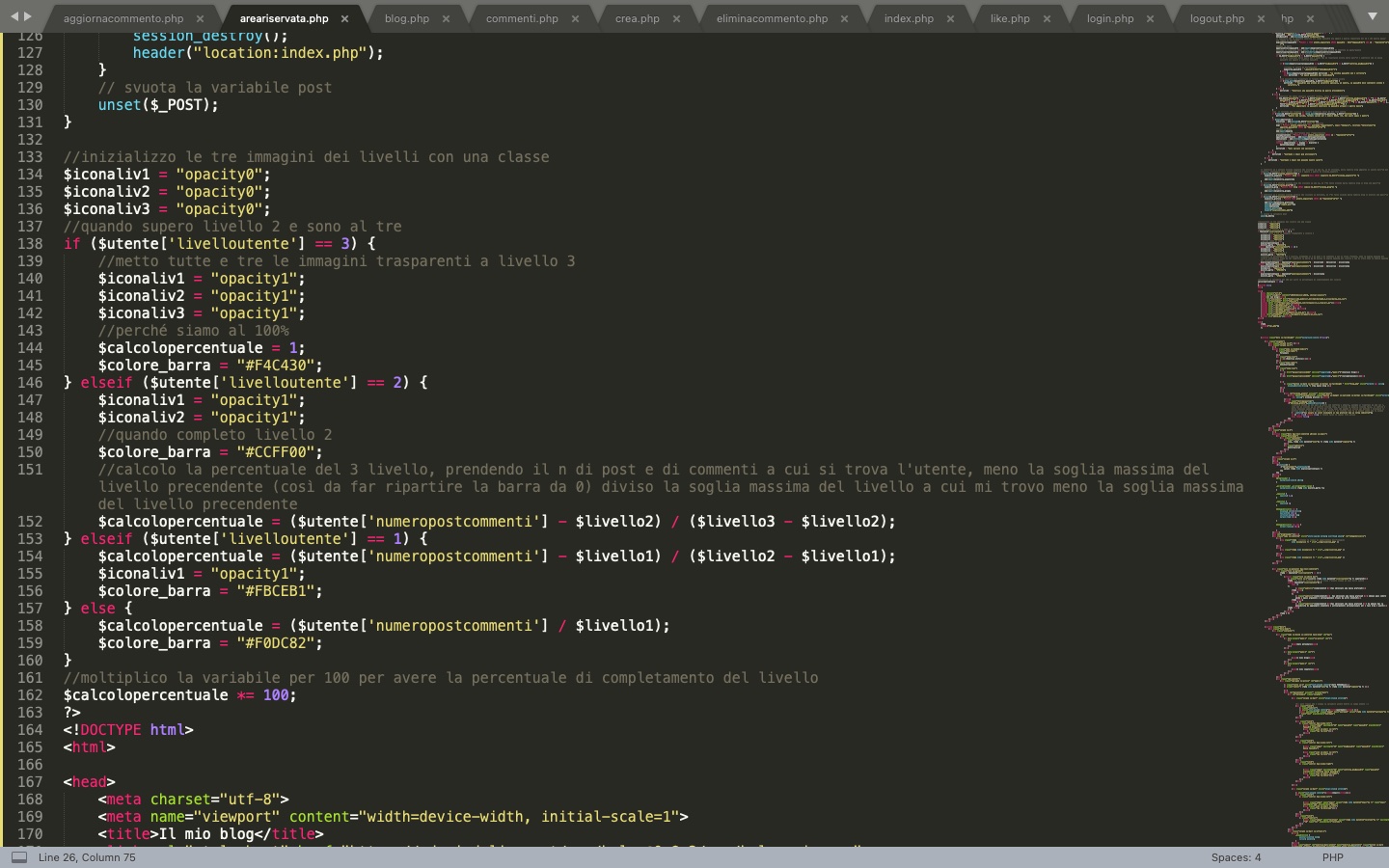
**AREA RISERVATA**

L’utente loggato al sito può accedere ad un’area riservata.

L’area riservata presenta, in alto, una barra che mostra lo stato di avanzamento dell’utente fra un livello e l’altro. La soglia dei vari livelli è definita nel file config.php, file nel quale si trova anche la funzione calcolalivelloutente(). La funzione recupera il numero di post e commenti inseriti da un utente, controlla a quale livello appartengono, e salva in sessione il livello dell’utente.

La funzione viene poi richiamata nel file areariservata.php, per ricalcolare il livello dell’utente e passargli l’id di quell’utente. Le immagini per i tre livelli sono invece assegnate ad una classe, che fa sì che inizialmente le immagini abbiano un’opacità e che, a mano a mano, che l’utente avanza di livello viene modificata e rende le immagini trasparenti. Nel file viene poi calcolata anche la percentuale dei vari livelli e questo valore viene poi passato alla funzione progressbar, che fa progredire la barra sulla base della percentuale di completamento del livello calcolata.

L’inizializzazione delle classi per le immagini e il calcolo della percentuale di avanzamento di livello:



L’area riservata è suddivisa in tre tab, che vengono gestite attraverso javascript e sono: dati personali, miei blog e i miei coautori.

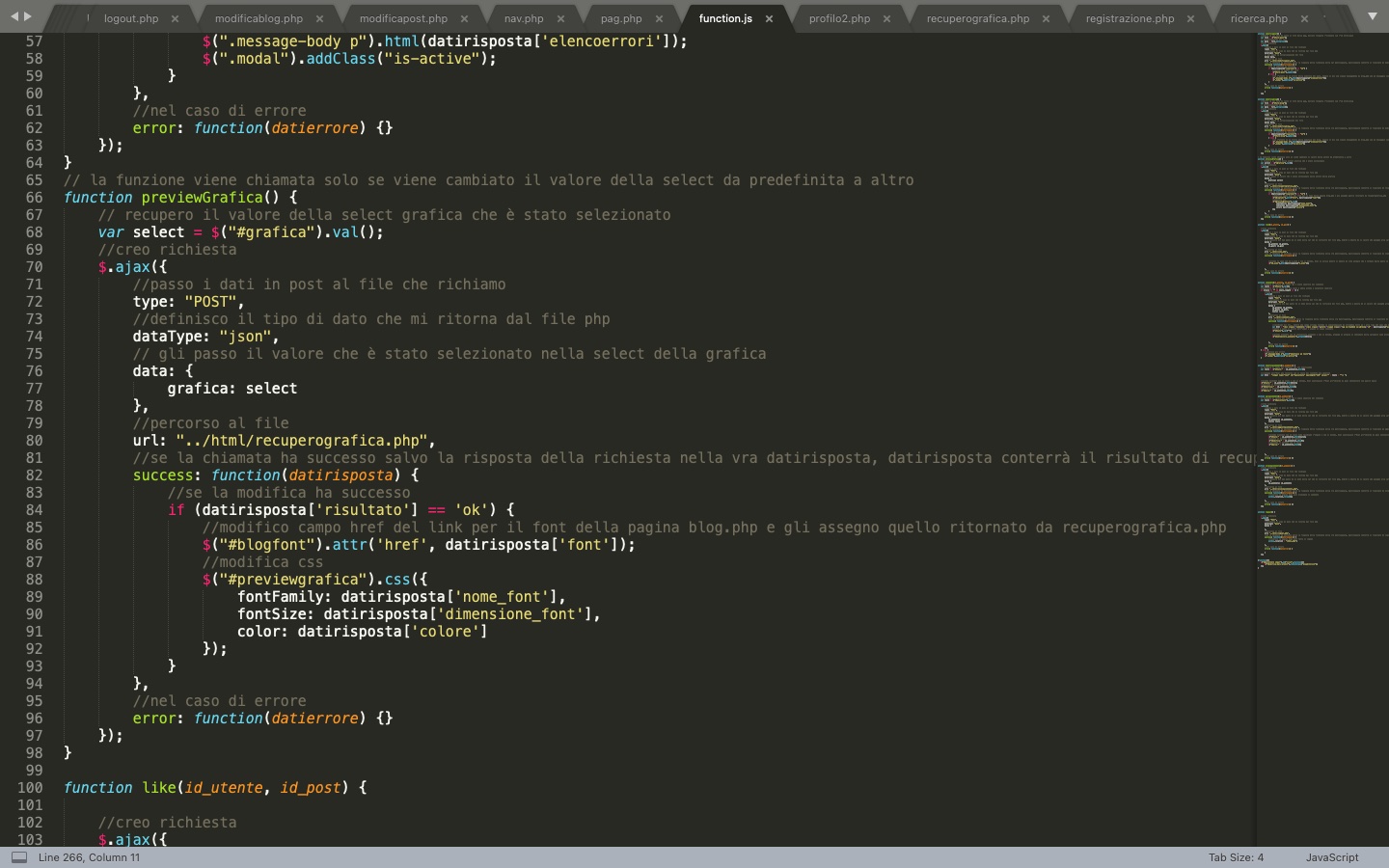
In dati personali l’utente può modificare i dati che aveva inserito precedentemente in fase di registrazione: username e email prendono di default il valore inserito e, in caso di modifica, si controlla che non siano già stati utilizzati, che rispettino un certo formato e che non siano vuoti quando l’utente clicca il pulsante per la modifica dei dati; il telefono, se inserito, assume quel valore, se modificato, invece, deve rispettare il formato prestabilito; per quanto riguarda le password, per modificare la password l’utente deve inserire la password attuale che deve coincidere con quella inserita in fase di registrazione, deve poi inserire una nuova password che rispetti il formato stabilito e confermarla nel campo conferma password: se uno di questi campi viene lasciato vuoto, comparirà un messaggio di avviso all’utente, altrimenti, comparirà un messaggio di conferma della modifica dei dati.

In i miei blog, l’utente può creare un nuovo blog, modificare o cancellarne uno esistente oppure aggiungere un post ad un blog già esistente. In questa sezione verranno aggiunti anche i blog per i quali l’utente è stato nominato coautore da altri utenti e che, però, non potrà eliminare.

In i miei coautori l’utente può invece visualizzare gli utenti che ha nominato come coautori e i blog per i quali sono stati nominati e, attraverso il pulsante elimina coautore, togliere loro la carica.

**CREA BLOG:**

Durante la fase di creazione del blog, l’utente deve inserire alcuni campi obbligatori: titolo, che deve essere lungo almeno 3 caratteri e non più di 25, e per cui si controlla, inoltre, che non esista già un blog con quel titolo nel database; descrizione, campo che va dai 20 ai 50 caratteri; argomento: gli argomenti vengono recuperati con una query e ciclati con un foreach per visualizzarli come valori della select, quando l’utente raggiunge il livello due si mostrano anche i nuovi argomenti che sono stati inseriti da altri utenti che hanno già completato il terzo livello; grafica: l’utente può selezionare una grafica, se non la seleziona verrà adottata una grafica di default per il blog, le grafiche vengono recuperate con una query sulla tabella grafica filtrando per il livello dell’utente, così da visualizzare in pagina le grafiche disponibili per l’utente sulla base del livello a cui si trova e poi vengono ciclate con un foreach per inserirle come valori della select. Una volta selezionata una grafica, è possibile vedere lo stile in una sezione apposita che viene gestita con Ajax dalla funzione previewgrafica().

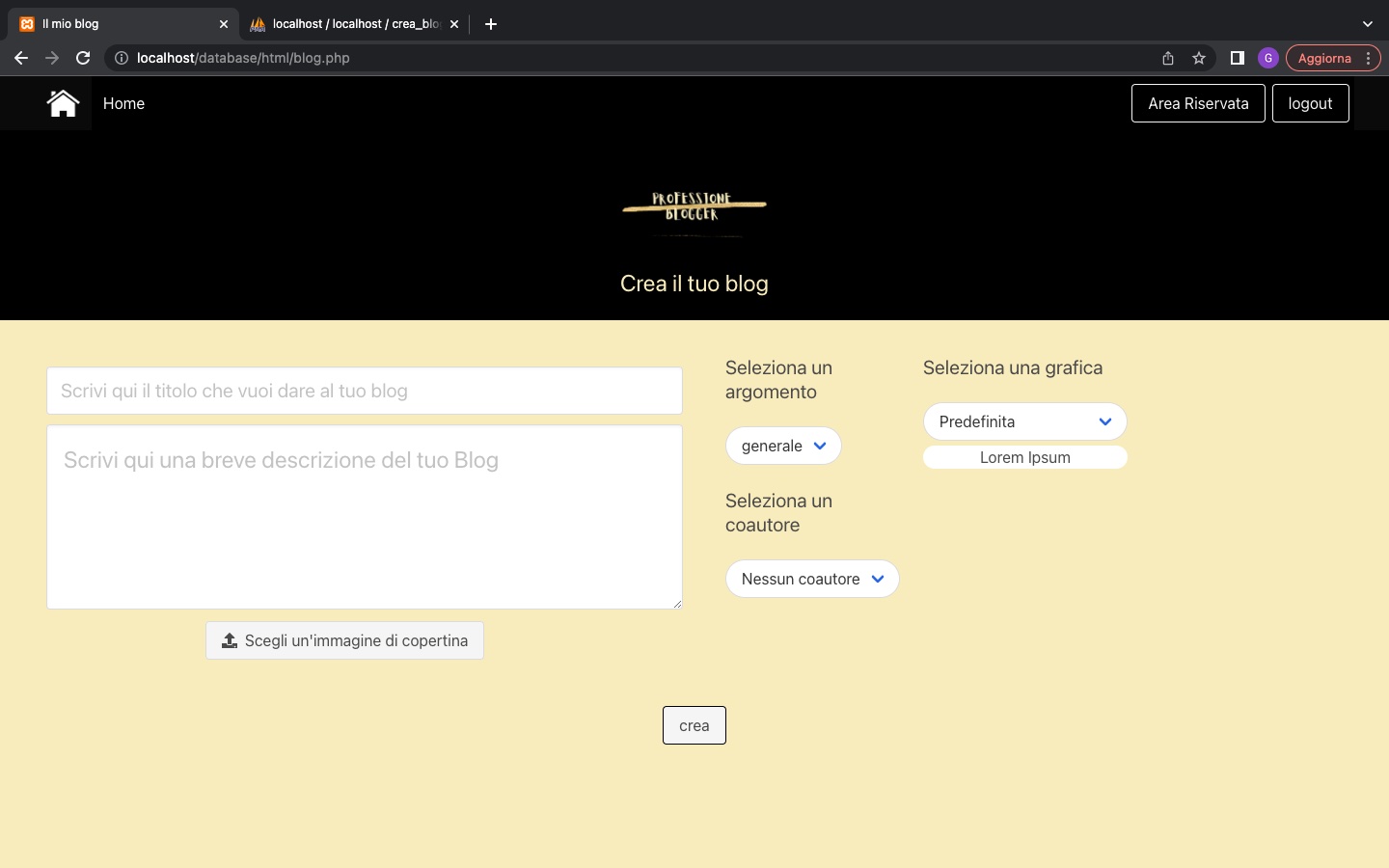


Attraverso l’attributo “onchange” inserito nella select, la funzione previewgrafica() si attiva soltanto quando viene cambiato il valore della select (da predefinito ad altro). La funzione recupera il valore selezionato nella select, avvia la richiesta ajax chiamando il file recuperografica.php, nel quale faccio il recupero dell’id della grafica selezionata e lo salvo, se questo processo va a buon fine, allora, la funzione previewgrafica() modifica il link con il font, inserendo quello ritornatogli da recuperografica.php e modifica il css del div in cui si mostra l’anteprima della grafica selezionata.

In caso di completamento del terzo livello, si rendono visibili nuove grafiche e si aggiunge il campo per inserire un nuovo argomento.

L’utente può scegliere di aggiungere un coautore per il suo blog, selezionandolo dalla select apposita: anche in questo caso, con una query sulla tabella utente\_registrato recupero gli utenti registrati e, quindi, nominabili come coautori, e poi ,ciclandoli con un foreach, li inserisco come valori della select. L’ultimo campo (opzionale) riguarda la selezione di un’immagine di copertina per il blog: le immagini vengono gestite ciclando la variabile $\_FILES, controllando che il file sia stato inserito (no vuoto), recuperando le informazioni dell’immagine e salvandola con l’id del blog, il suo nome e la sua estensione per poi stabilire il percorso del file e attraverso la funzione move\_uploaded\_file() spostarla nella cartella di destinazione.

Quando l’utente clicca sul pulsante crea viene eseguita la submit del form e, se la variabile errori è vuota, vengono inseriti i nuovi valori nel database e viene fatto il redirect alla pagina ricercablog.php.

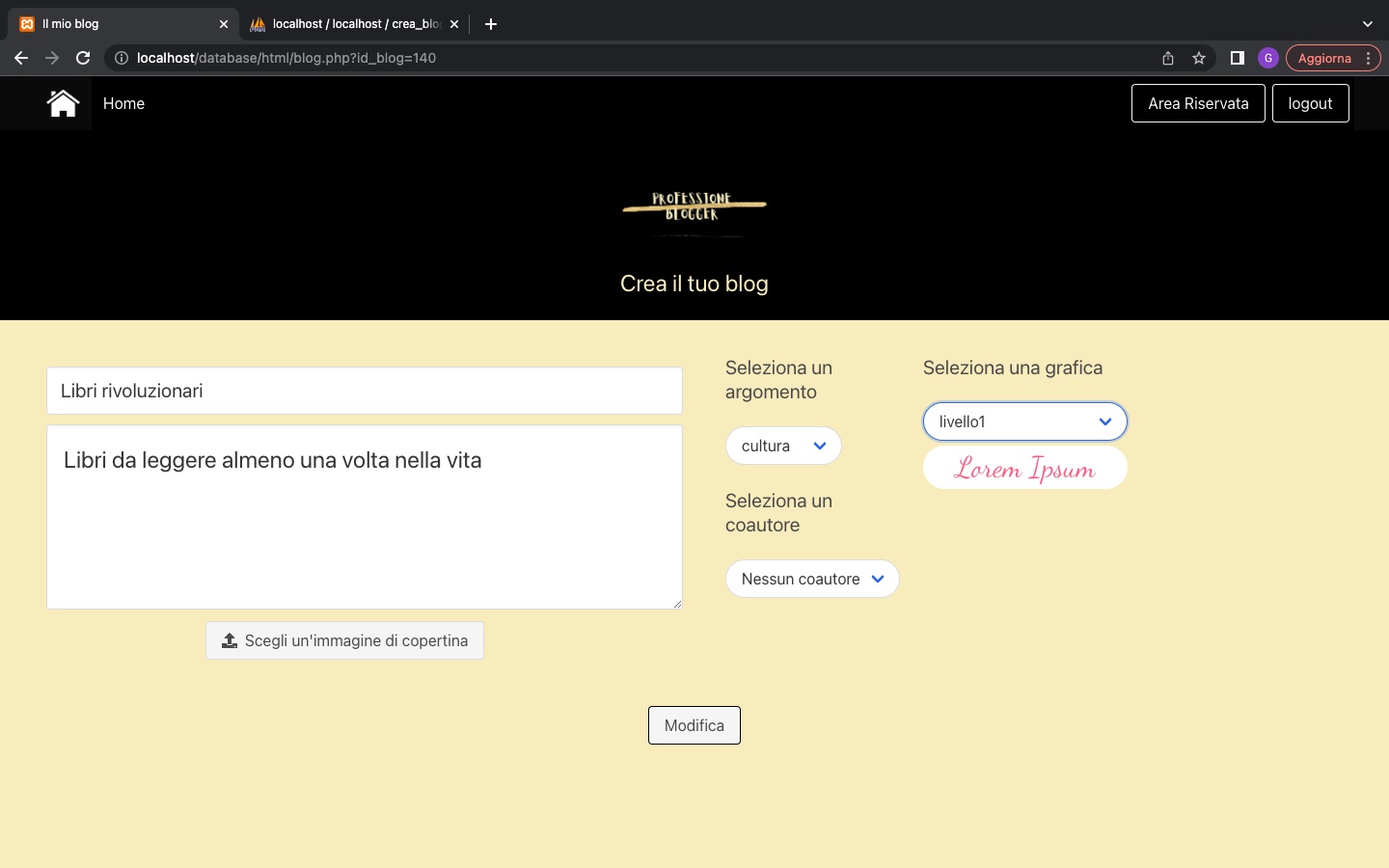


**MODIFICA BLOG**

Sempre nella pagina blog.php controllo all’inizio della pagina se la variabile id\_blog è settata: se è settata, il blog esiste già e faccio la get per recuperarne i dati, altrimenti ritorno false e significa che si tratterà di un blog nuovo.

Infatti, se l’id blog è uguale a false farò gli inserimenti nel database e farò comparire all’utente il pulsante crea nuovo blog, altrimenti, recupererò con una query i dati del blog e farò comparire il bottone modifica blog. La modifica del blog viene gestita con Ajax, attraverso la funzione modificablog(): si recupera il form del blog, si avvia una richiesta ajax, chiamando il file modificablog.php. In modificablog.php si controlla che la var id\_blog sia settata e si fanno i controlli sulla modifica dei campi titolo e descrizione. Per i campi grafica e coautore nella pagina blog.php, viene preimpostata la select in modo tale da mostrare, in fase di modifica, il coautore/grafica che erano stati selezionati precedentemente in fase di creazione del blog. In modificablog.php specificherò che se questi due campi non sono stati selezionati rimangono null, altrimenti prendono il nuovo valore selezionato dall’utente. Viene fatto poi un update dei dati del blog e si resitituisce il risultato alla funzione modificablog(). Se la modifica è andata a buon fine la funzione farà il submit del form nella pagina blog.php in cui verrà fatto il redirect a ricercablog.php. Altrimenti, se l’utente fa un errore nella modifica del blog si stamperanno i messaggi di errori ritornati da modificablog.php popolando il div presente in blog.php.

Per la modifica delle immagini, si veriica in blog.php che l’utente stia modificando il blog e, in tal caso, quando l’utente carica un file, si elimina il file preesistente e si sostituisce con quello nuovo.

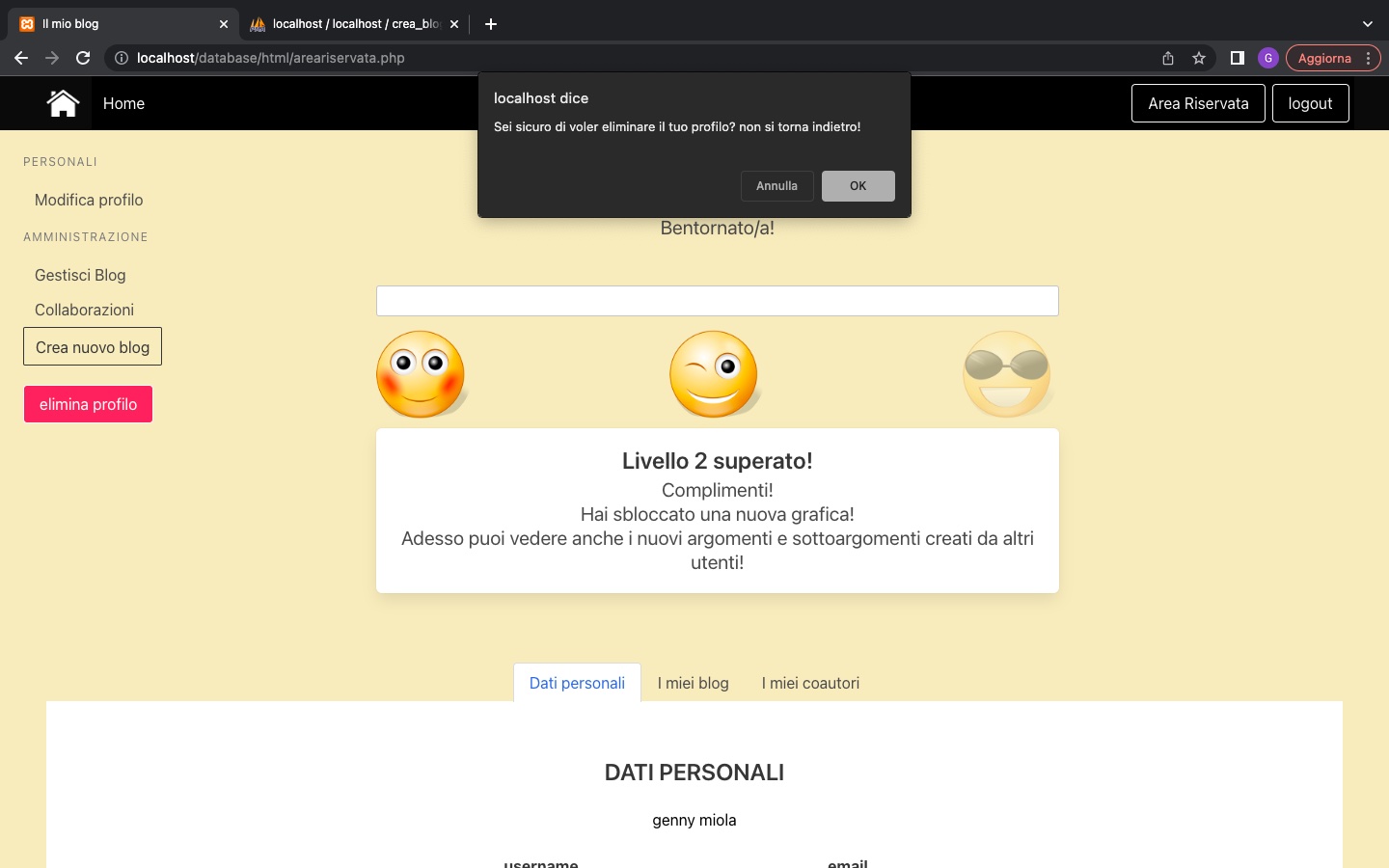


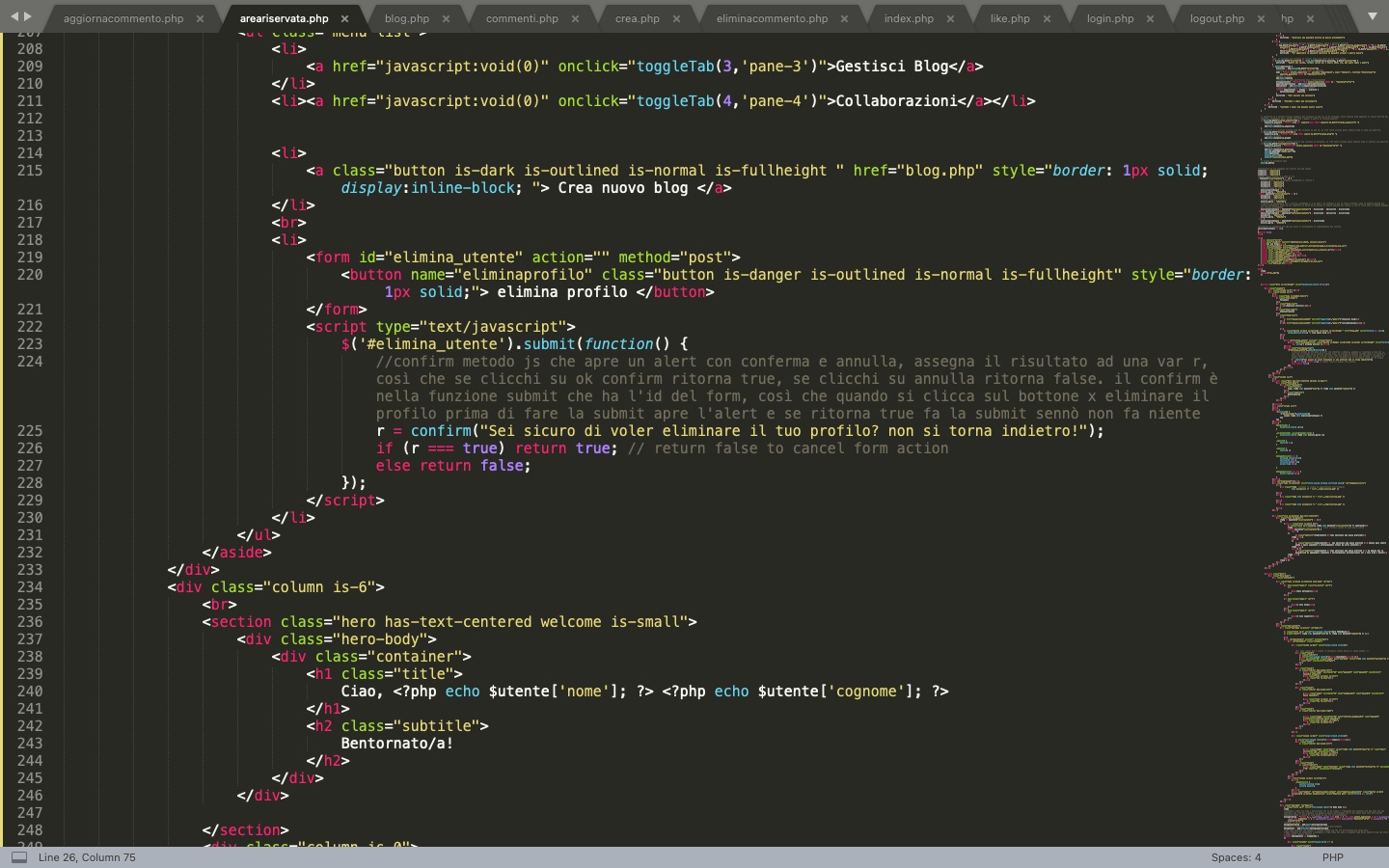
**ELIMINA BLOG**

Nella pagina areariservata.php è presente il pulsante elimina blog, se l’utente clicca sul pulsante viene eseguita una query che andrà ad eliminare dalla tabella blog il blog con quell’id.

**ELIMINA PROFILO**

Nella pagina areariservata.php è presente il pulsante elimina profilo, su l’utente clicca sul pulsante, viene prima avviata una funzione javascript che permette di aprire un alert in cui l’utente può confermare o annullare la propria decisione, nel caso di conferma viene eseguita una query che elimina dalla tabella utente\_registrato il profilo con quell’id, viene poi svuotata la sessione e la variabile contenente i dati dell’utente, distrutta la sessione e l’utente viene reindirizzato alla index.





**ELIMINA COAUTORE:**

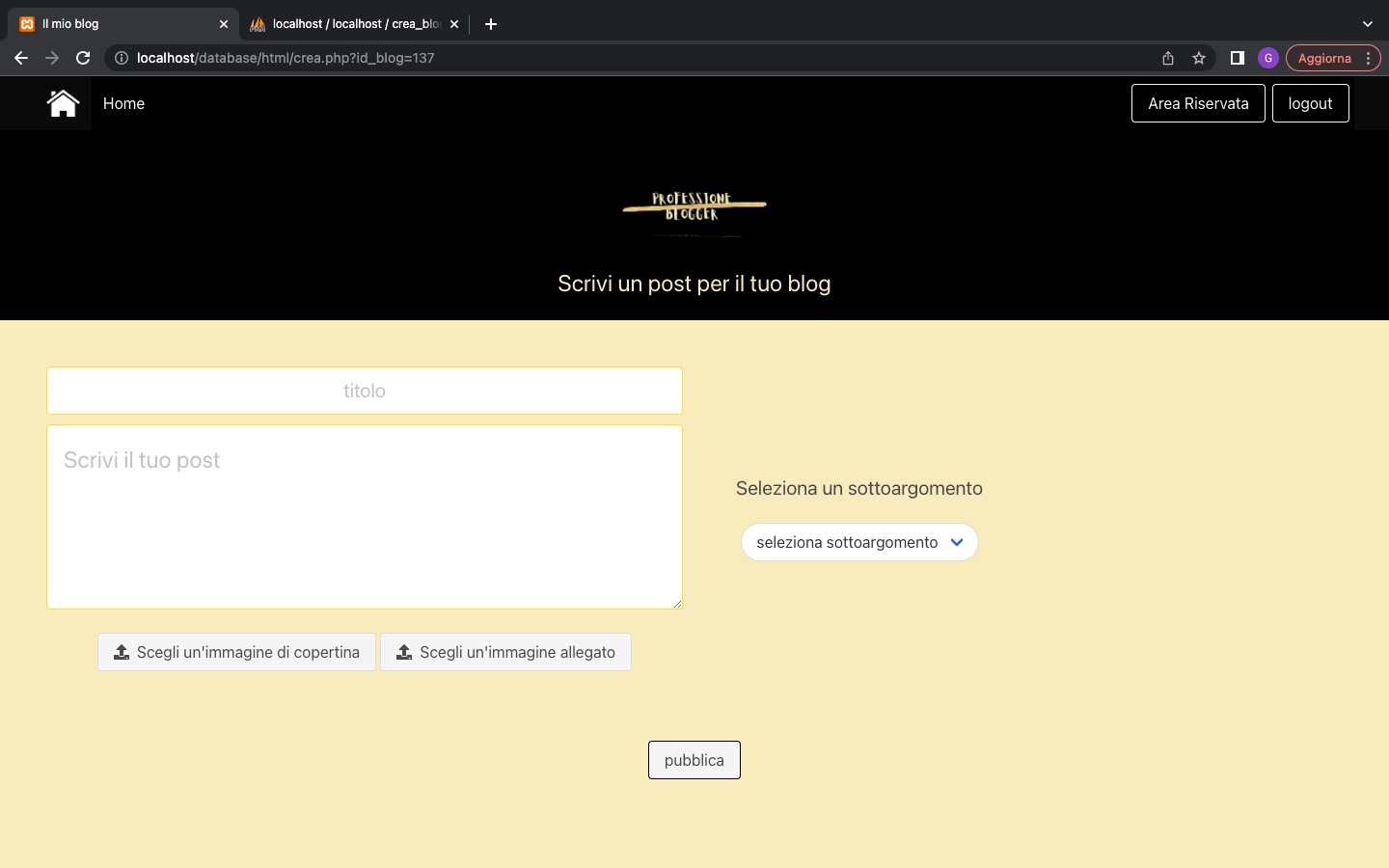
Nella pagina areariservata.php è presente il pulsante elimina coautore, su l’utente clicca sul pulsante viene eseguita una query che aggiorna nella tabella blog il valore dell’id del coautore impostandolo a null.

**CREA POST:**

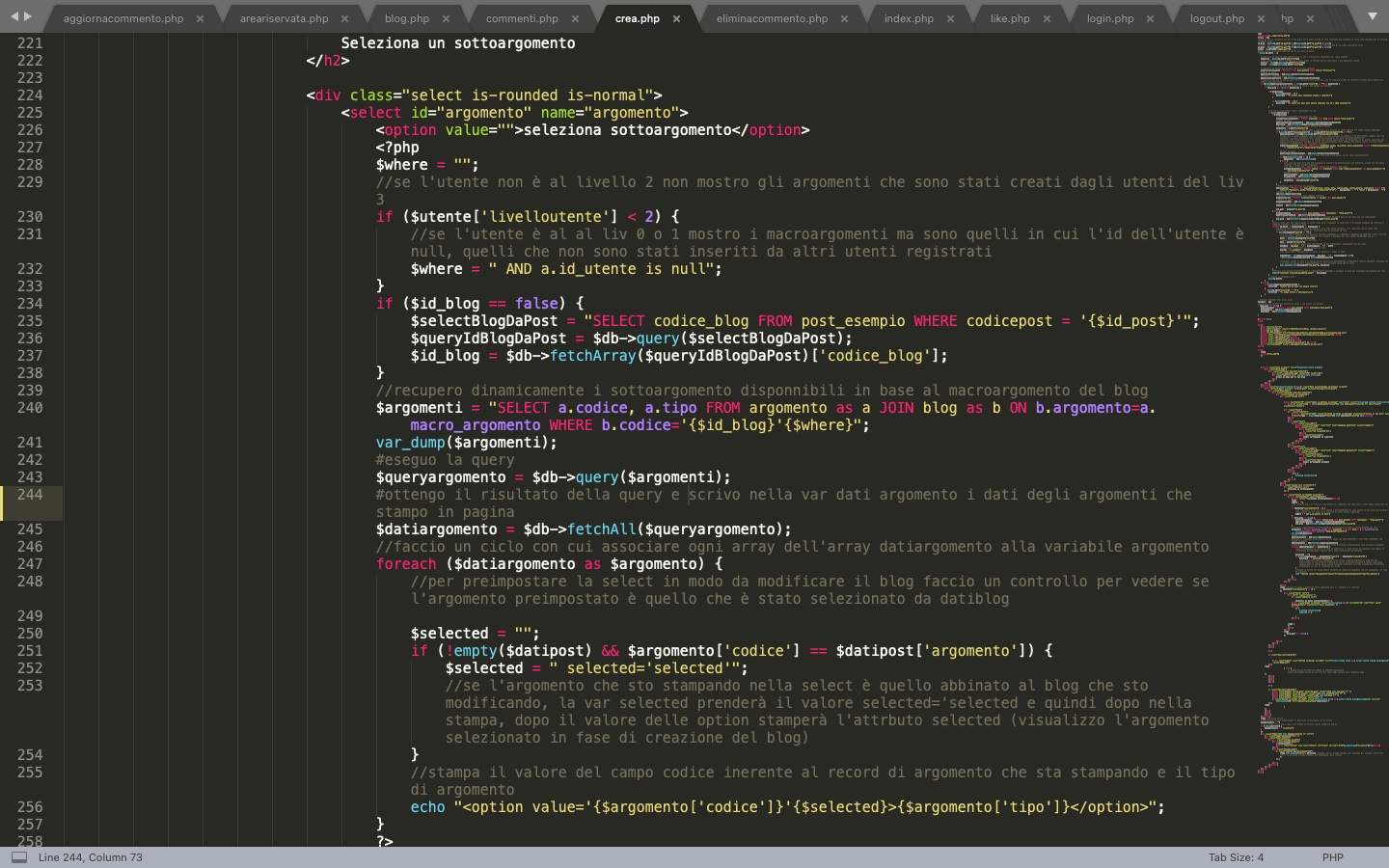
Nella fase di creazione del post, l’utente deve compilare alcuni campi obbligatori: il titolo è un campo con lunghezza minima 3 caratteri e massima 50; testo è un campo che ha una lunghezza minima di 50 caratteri e massima di 2000. Ad entrambi è applicata la funzione escape.

I campi opzionali riguardano la possibilità di selezionare un’immagine di copertina e/o un’immagine allegato, usando lo stesso metodo sfruttato per l’inserimento di un’immagine nel blog e la selezione di un sottoargomento. Per i sottoargomenti, uso una select che recupera i sottoargomenti disponibili sulla base del macroargomento selezionato dall’utente per il blog. Quando l’utente raggiunge il livello due vengono mostrati anche i nuovi sottoargomenti che sono stati inseriti da altri utenti che hanno già completato il terzo livello. Se poi l’utente si trova al terzo livello viene aggiunto un campo che gli permette di inserire un nuovo sottoargomento.

Quando l’utente clicca sul pulsante crea viene eseguita la submit del form e, se la variabile errori è vuota, vengono inseriti i nuovi valori nel database e viene fatto il redirect alla pagina ricercablog.php.



Il recupero dei sottoargomenti nella pagina di creazione del post:



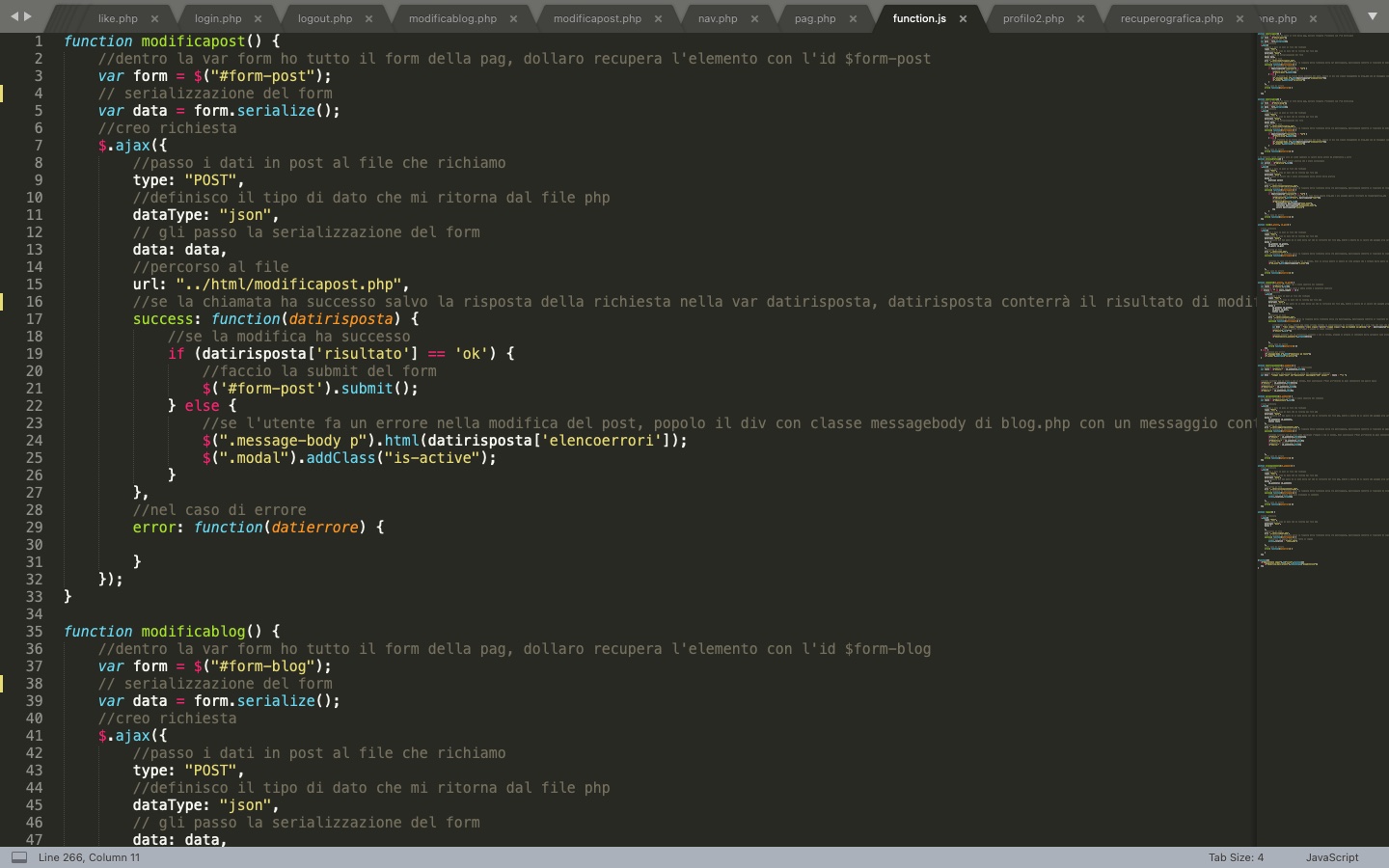
**MODIFICA POST:**

Dalla pagina di visualizzazione del post, pag.php, per l’utente autore del post, è possibile modificare il post attraverso il pulsante modifica post.

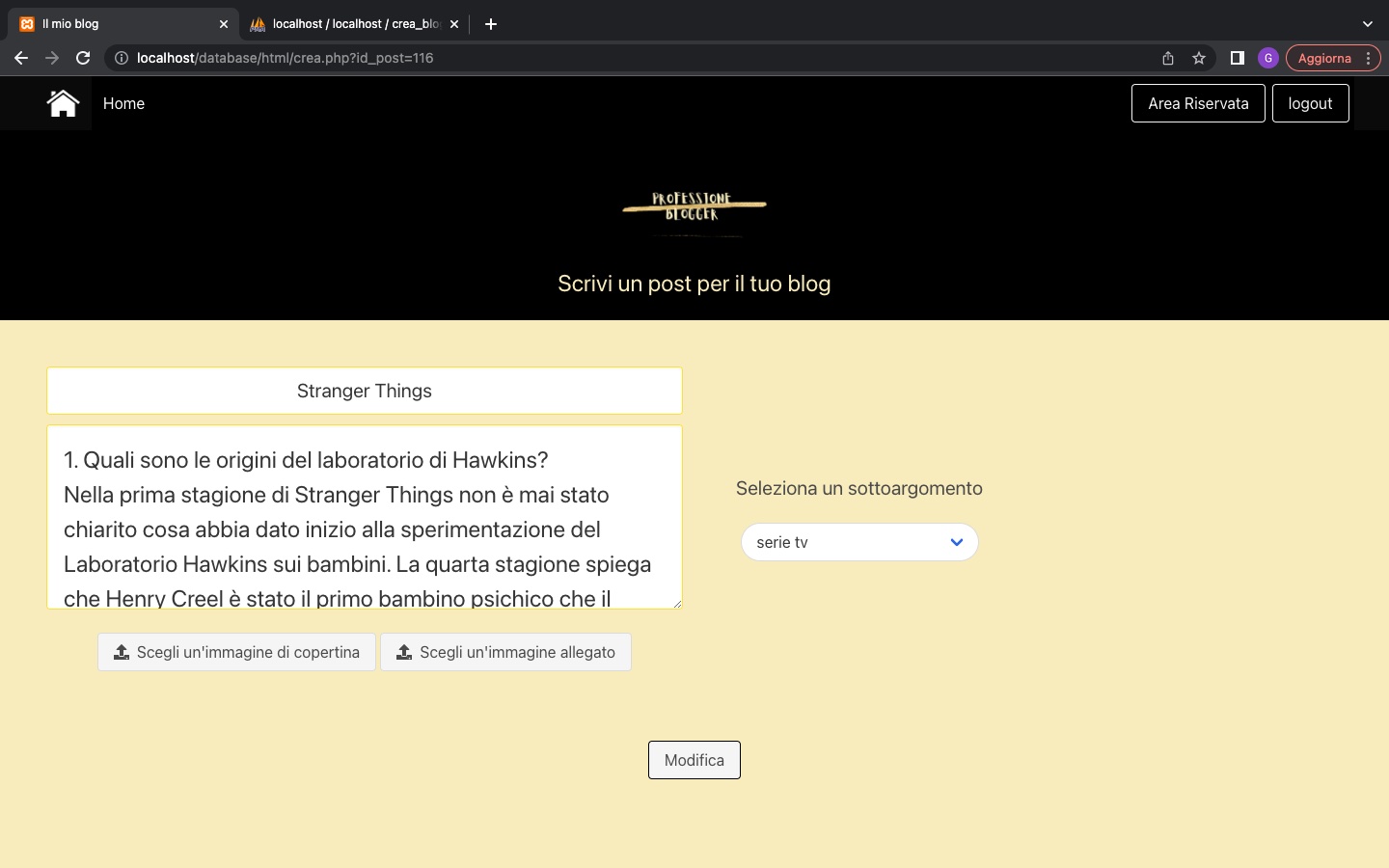
La modifica del post, così come quella del blog, avviene in ajax: si ha la funzione modificapost(), che recupera il form del post presenta in pag.php, avvia la richiesta ajax e chiama il file modificapost.php. In modificapost.php si verifica che l’id del post sia settato, in tal caso, si fanno i controlli sui campi titolo e descrizione, così come per l’inserimento, anche per la modifica. In crea.php viene impostata la select di sottoargomento in modo tale da preimpostarla con il sottoargomento che era stato selezionato precedentemente dall’utente in fase di creazione del post. In modificapost.php, se non ci sono errori, si aggiornano i dati del database con i nuovi valori inseriti e si restituisce il risultato alla funzione modificapost(). Se la modifica è andata a buon fine la funzione farà il submit del form in crea.php e verrà fatto il redirect a pag.php, mostrando il post modificato, altrimenti, se l’utente fa un errore nella modifica del post, verranno stampati i messaggi di errori ritornati da modificapost.php, popolando il div presente in crea.php.

Per la modifica delle immagini, si verifica in crea.php che l’utente stia modificando il post e, in tal caso, quando l’utente carica un file, viene eliminato il file preesistente e caricato quello nuovo.

La modifica del post in Ajax:



Modifica del post lato user:



**ELIMINA POST:**

Dalla pagina ricercablog.php l’utente registrato e proprietario del post può, cliccando sul pulsante, elimina post, cancellare il post creato. In tal caso, viene avviata una query di delete che cancella dalla tabella post\_esempio il post con quell’id. Nel caso di utenti coautori di un blog, questi potranno cancellare soltanto i post che hanno creato loro per quel blog, ma non quelli creati dall’utente proprietario del blog.

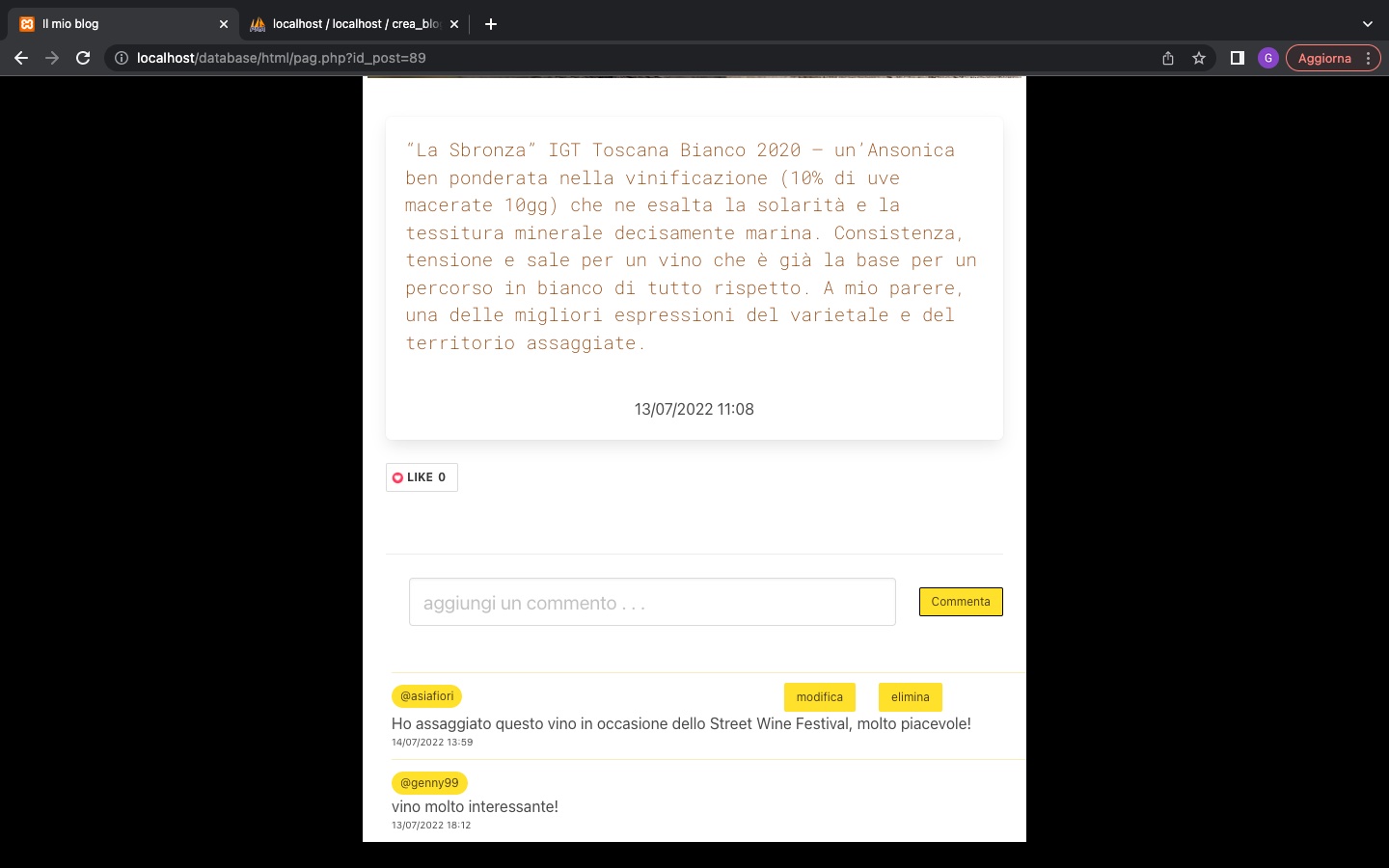
**COMMENTI:**

L’utente registrato può inserire un commento, modificarlo e eliminarlo, mentre un utente non registrato o loggato può soltanto vedere i commenti pubblicati da altri utenti.

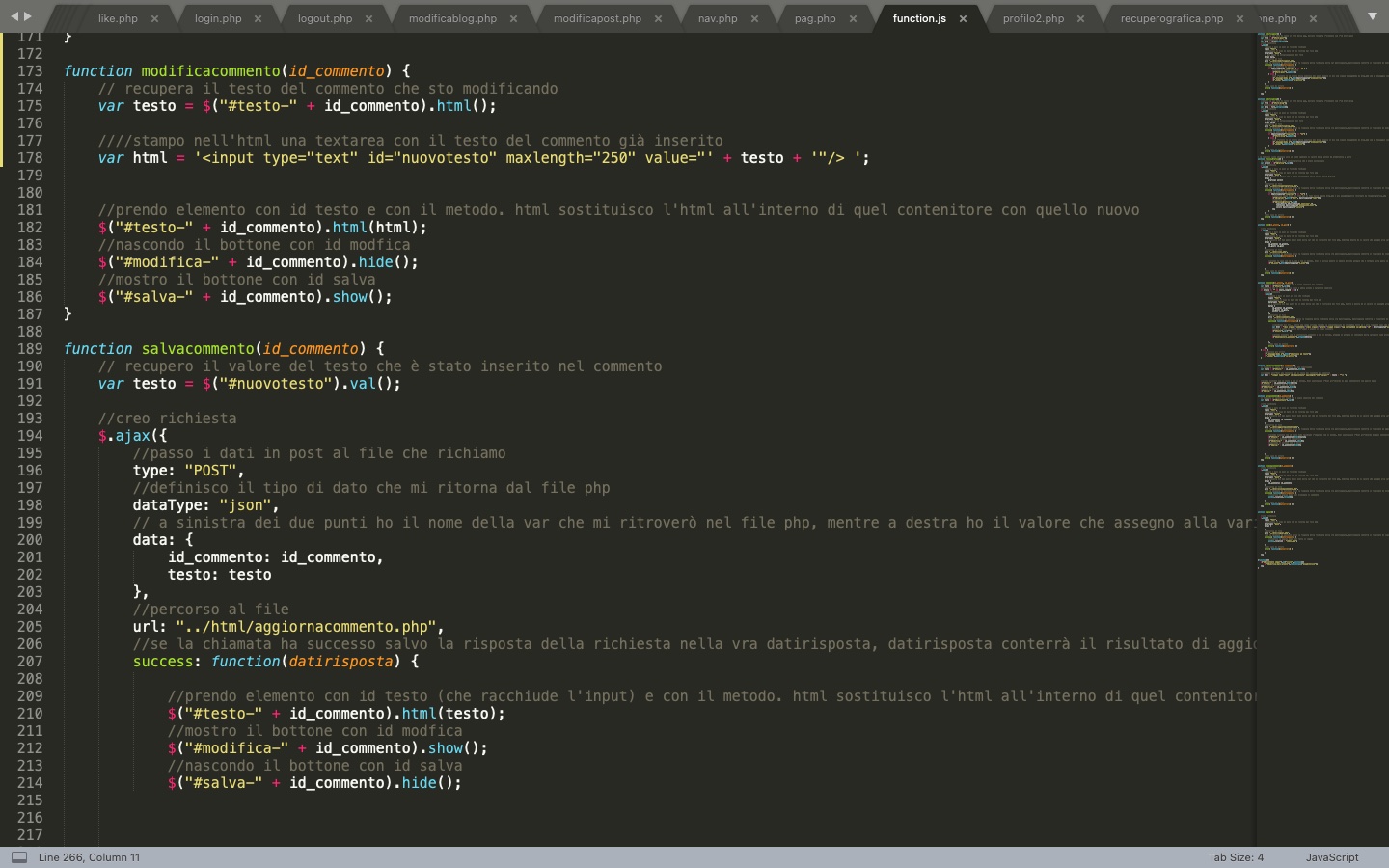
I commenti vengono gestiti in Ajax con la funzione commento() e sono visualizzabili nella pagina del post pag.php. In pag.php recupero tutti commenti inseriti attraverso una query che filtra per il codice del post e li ordina dal più al meno recente. Ciclo poi i commenti con un foreach e li stampo in un div, insieme allo username dell’utente che li ha pubblicati e alla data e ora di pubblicazione. Per inserire un nuovo commento l’utente loggato clicca sul pulsante commenta, dopo aver inserito un testo, e viene chiamata la funzione commento(): la funzione recupera il valore del testo inserito nel commento, controlla che il testo non sia vuoto e apre una richiesta ajax chiamando il file commenti.php a cui passa l’id dell’utente, l’id del post e il testo. In commenti.php, dopo aver fatto controlli che il testo non sia vuoto, si inseriscono i nuovi valori nel database. Attraverso la funzione lastinsertid() recupero poi l’id dell’ultimo commento inserito, per ordinarlo dal più recente al più vecchio e con una query recupero anche lo username dell’utente che lo ha inserito e la data di inserimento. Se l’inserimento ha successo, la funzione commento() aggiunge testo, username a data del commento al div in pag.php e poi setta a vuoto il valore dell’input, svuotando il campo del commento.

L’utente che ha inserito un commento può modificarlo o eliminarlo. La modifica del commento avviene in Ajax: in pag.php viene fatto un controllo che l’utente che visualizza la pagina sia loggato e sia quello che inserito il commento. Se l’utente clicca sul pulsante modifica commento viene chiamata la funzione modificacommento() che recupera il testo del commento che l’utente sta modificando stampandolo in una textarea e sostituisce il vecchio testo con quello nuovo, nascondendo il bottone modifica e mostrando il bottone salva, che prima era in display none. Se l’utente clicca su salva viene chiamata la funzione salvacommento(), in cui si recupera il valore del testo inserito e si apre una richiesta ajax chimando il file aggiornacommento.php e passandogli l’id del commento e il testo. In aggiornacommento.php controllo di non inserire commenti vuoti e, in tal caso, faccio la insert dei nuovi valori nel database. Se la modifica va a buon fine la funzione salvacommento() sostituisce il testo nuovo, lo salva e nasconde il bottone salva, mostrando di nuovo quello di modifica.

L’utente può anche decidere di eliminare il commento. In questo caso, quando preme sul bottone elimina commento viene chiamata la funzione eliminacommento() che crea una richiesta ajax alla pagina eliminacommento.php, in cui viene eliminato con una delete il commento con quell’id. Se l’eliminazione va a buon fine, viene ricaricata la pagina.



Le funzioni di modifica e salva commento in Ajax:



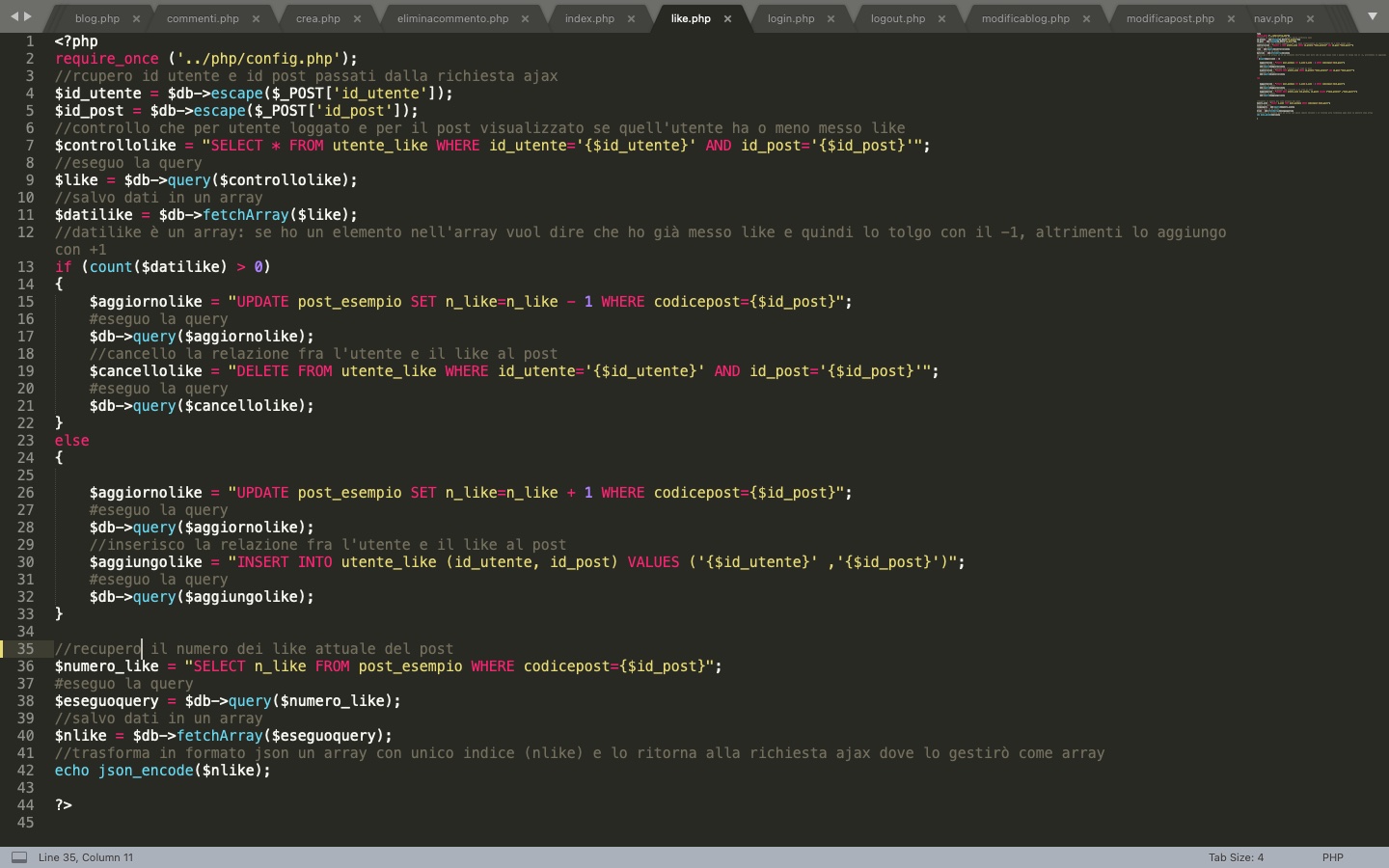
**LIKE:**

L’utente registrato può inserire e togliere un like, mentre un utente non registrato o loggato può soltanto vedere il numero di like ricevuti per ciascun post.

In pag.php viene fatto il controllo se l’utente che sta visualizzando la pagina è loggato o meno: se non lo è, stampo il numero di like per quel post, altrimenti mostro il bottone per far interagire l’utente con il post. I like sono gestiti in ajax, quando l’utente clicca su pulsante like viene chiamata la funzione like() che apre una richiesta ajax e chiama la pagina like.php, passandogli l’id dell’utente e del post. In like.php controllo che per quell’id utente e per quell’id del post se l’utente abbia o meno messo like, attraverso una query e salvo il risultato in un array. Se nell’array ho un elemento significa che l’utente ha già messo like e, quindi, se clicca nuovamente sul bottone decremento il n\_like di 1, andando poi a cancellare la relazione fra l’utente e il like al post; altrimenti, aumento il n\_like di 1, aggiungendo la relazione fra l’utente e il like al post.

Poi, con una query recupero il numero di like del post e passo il risultato alla funzione like() che andrà a stampare in pagina il n di like per quel post.

Funzionamento della pagina like.php:



**LOGOUT:**

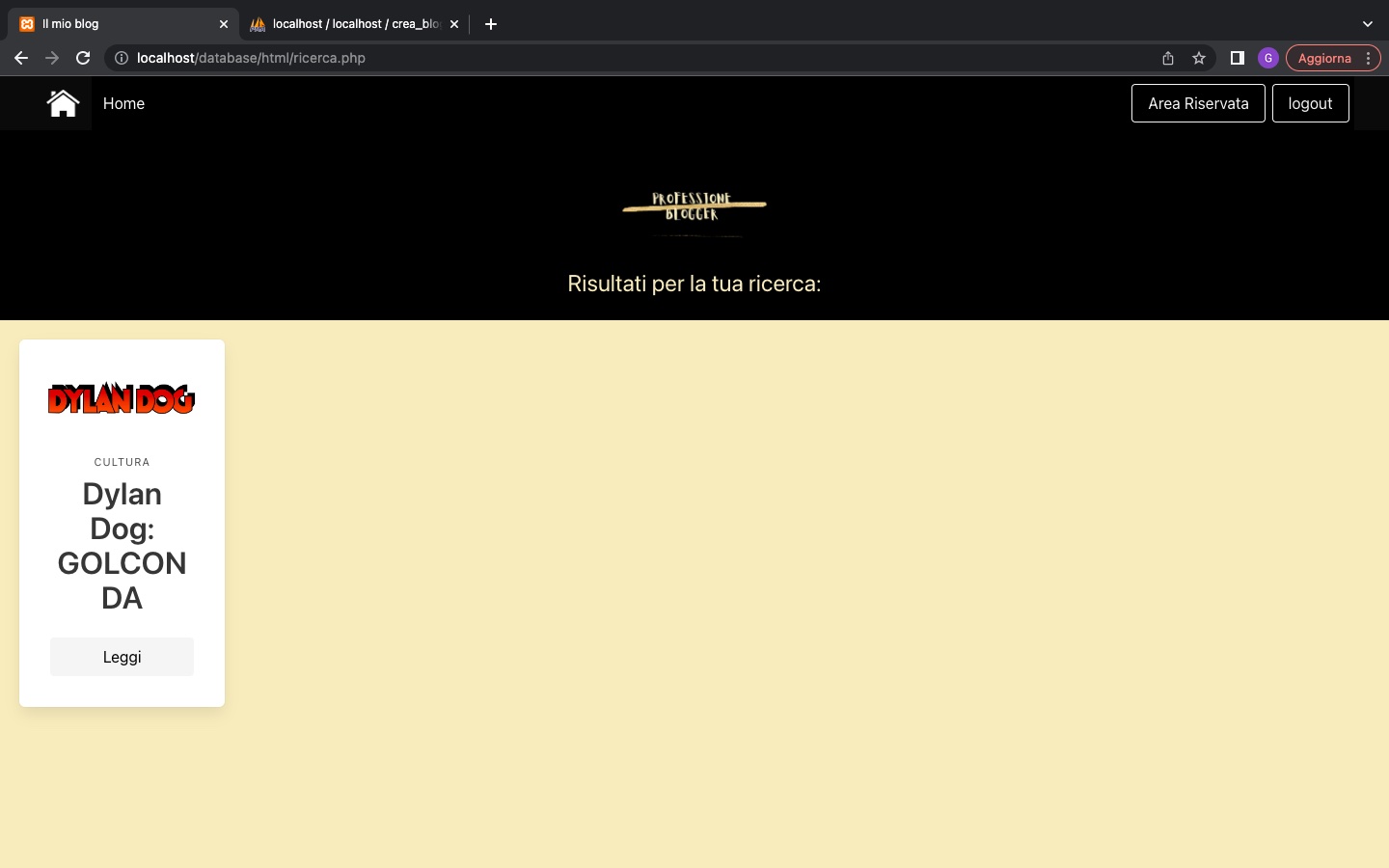
Il logout viene gestito in ajax. Quando l’utente clicca sul pulsante logout viene chiamata la funzione logout() che apre una richiesta ajax e chiama il file logout.php. Il file logout.php se trova una sessione già attiva riprende quella e poi svuota la sessione e slogga l’utente. Il risultato viene poi passato alla funzione logout() che fa il redirect alla pagina index.php.

**RICERCA:**

Dalla pagina index.php è possibile per qualsiasi utente ricercare un post o un blog attraverso due tipi di ricerca: per argomento o per titolo.

In index.php con una query recupero tutti i dati della tabella argomento e, ciclandoli con un foreach, li stampo come valori di una select. Quando l’utente preme su ricerca, viene fatta la chiamata alla pagina ricerca.php, in cui se è settata la variabile $\_POST[‘genere], con una query si recuperano i blog che hanno come valore dell’argomento quello selezionato dall’utente. Se ci sono blog per quell’argomento, questi vengono stampati in dei box, assieme al titolo del blog, ad un’immagine di copertina (se presente) e insieme al nome dell’argomento di appartenenza.

L’utente può anche cercare un post oppure un blog digitando il titolo o parte di esso. Quando l’utente clicca su cerca, viene chiamata la pagina ricerca.php, nella quale, se è settata la variabile $\_POST[‘contenuto’] si recuperano con una query i blog o i post che contengono nel titolo ciò che è stato digitato dall’utente e viene fatta anche una join con argomento, blog e grafica. Se ci sono blog o post con quel titolo, questi vengono stampati in dei box, assieme al titolo del blog, ad un’immagine di copertina (se presente) e insieme al nome dell’argomento di appartenenza.



**SICUREZZA:**

Per la sicurezza del sito utilizzo la funzione di escape, che usa il metodo mysqli\_real\_escape\_string() che sostituisce apice, doppio apice, backslash e stringa nulla con le rispettive sequenze di escape in modo tale da evitare eventuali attacchi; la funzione htmlspecialchars() , così da convertire i caratteri speciali di una stringa e trasformarli nei corrispondenti codici HTML.

Controllo poi la lunghezza dei vari campi di input, per la loro dimensione minima e massima.

Per gli input di tipo file, controllo con l’attributo “accept” che i file caricati siano immagini.

Uso la funzione trim() per eliminare gli spazi bianchi e far sì che i campi non restino vuoti.

Per la codifica delle password utilizzo MD5 e per ulteriori controlli di sicurezza dei campi riguardanti password, email e telefono utilizzo la funzione pregmatch(), stabilendo dei formati validi in espressioni regolari e controllando, poi, che ciò che viene inserito dall’utente nell’input corrisponda al formato richiesto.

