

# Gestione di un Forum

## Indice generale

Analisi .....	1
Schema ER.....	2
Dizionario dei dati .....	3
Schema logico .....	3
Scelte implementative .....	4
DataBase utilizzato .....	8
SQL .....	11
Manuale d'uso .....	12

## Analisi

### Problema generale:

Deve essere realizzato un database dal nome “Forum”, che dovrà contenere tutte le *informazioni* relative alle attività che normalmente sono offerte dai forum in Internet.

Quindi, dovranno essere memorizzate *informazioni* suddivise in **argomenti** relativi ad un'unica **tematica** del forum (“Cinema”) prevedendo che per ogni argomento si possano inviare messaggi che costituiscono l'inizio di una discussione che si arricchisce di contributi o *post* (risposte al messaggio). Come in ogni forum, è previsto che per poter partecipare attivamente alle funzionalità fornite (inserire un nuovo argomento, scrivere messaggi di inizio discussione oppure post), ogni utente sia costretto a registrarsi; mentre il contenuto informativo (le discussioni o **threads**) sarà reperibile da qualsiasi utente navighi sul web.

Si rende quindi necessaria la memorizzazione dei dati dell'utente che intende registrarsi, ossia il nickname e la password ed eventualmente altri dati opzionali quali ad esempio la data di registrazione e la località di provenienza, inoltre si memorizza anche il numero dei messaggi che l'utente ha inviato dopo la registrazione. In questo modo tutti gli utenti registrati potranno aprire un nuovo argomento e scrivere/modificare un post.

Come facilmente intuibile il forum è un oggetto reperibile solo on-line, perciò come tale, non avrà l'aspetto di un' applicazione, ma bensì di una pagina web. Grazie a pagine dinamiche, dovrà essere possibile per l'utente registrato accedere al database, per poter inserire, modificare, o cancellare le informazioni contenute all'interno del forum. Sempre tramite pagine dinamiche verrà visualizzato, a chiunque lo desideri, il contenuto del database cioè le discussioni classificate per argomento.

Ovviamente tali modifiche e visualizzazioni si intendono *lato client*, poiché il *lato server* è gestito unicamente dall'amministratore, che è l'unico che può eliminare eventuali messaggi d'inizio discussione e l'unico che può apportare modifiche significative alla struttura del database.

## Obiettivi realmente raggiunti:

Sono stati realmente raggiunti tutti gli obiettivi prefissati (pur se la gestione dei permessi di accesso non è stata completata diversificando - a livello della struttura del DB - tra utente qualsiasi, utente registrato, figura **moderatore**) e sono state aggiunte ulteriori funzionalità.

Riguardo alle informazioni relative agli argomenti del forum, è stata aggiunta la visualizzazione dei dati relativi all'ultima risposta per quel dato argomento (ossia la data e l'ora dell'ultima modifica), il numero di risposte ricevute e il nome dell'autore che l'ha creato.

Per ogni messaggio, relativo ad un argomento, oltre a visualizzare il testo, è stata aggiunta la visualizzazione dei dati relativi l'ultima modifica del post (quindi la cronologia: data e ora), e quelli dell'autore. In particolare rispetto all'autore del messaggio vengono visualizzate le informazioni riguardanti l'ora e la data di registrazione, la località, e il numero di messaggi lasciati sul forum.

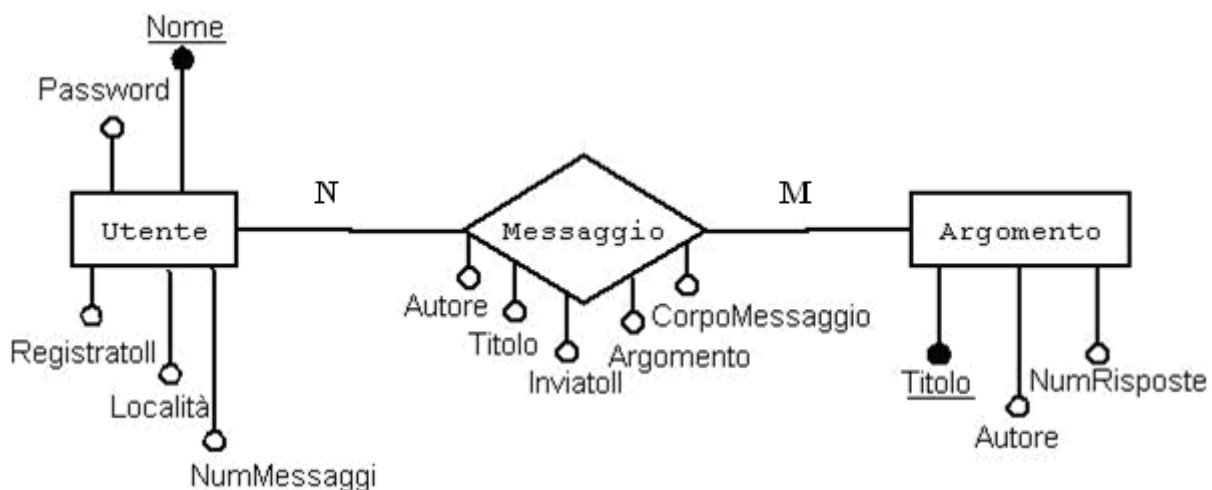
Con funzionalità anomala rispetto ai soliti forum, è possibile cancellare e modificare tutti i messaggi di ogni argomento (**post**), ad esclusione di quello relativo alla creazione dell'argomento stesso, che può essere solo modificato ma non cancellato.

Quindi un argomento, una volta creato, non può essere cancellato dal forum, e quindi dal database (questa operazione è consentita solo all'amministratore che aggiornerà opportunamente il DB).

Sono stati immessi dei controlli sulle attività relative al forum, **gestendo diversi permessi di accesso**. Ossia, ogni volta che si desidera effettuare qualche operazione di modifica sul database, è necessario fornire nome utente e password; con conseguente segnalazione di errore se i dati relativi all'utente sono errati. Nel caso, anomalo rispetto ai soliti forum, in cui si voglia cancellare o modificare un messaggio già esistente nel forum, non è necessario fornire il nome dell'utente, ma semplicemente la password poiché solo all'autore del messaggio è permesso modificarlo o cancellarlo (anche in fase successiva alla creazione).

Infine è stata realizzata una funzionalità che permette di cercare i messaggi all'interno del forum, tramite l'immissione del titolo o del testo del messaggio; con conseguente visualizzazione dei risultati della ricerca.

## Schema ER



# Dizionario dei dati

## Entità

Nome entità	Descrizione	Attributi	Identificatore
Utente	Gli utenti che si registrano nel forum	Nome, Password, RegistratoIl, Località, NumMessaggi	Nome
Argomento	Gli argomenti creati dagli utenti	Titolo, Autore, NumRisposte	Titolo

## Associazioni

Nome relazione	Descrizione	Entità componenti	Attributi
Messaggio	Gli utenti scrivono messaggi come creazione/risposte di argomenti	Utente (1,N), Argomento (1,N)	Autore, Titolo, CorpoMessaggio, InviatoIl, Argomento

## Schema logico

Argomento(Titolo, Autore, NumRisposte)

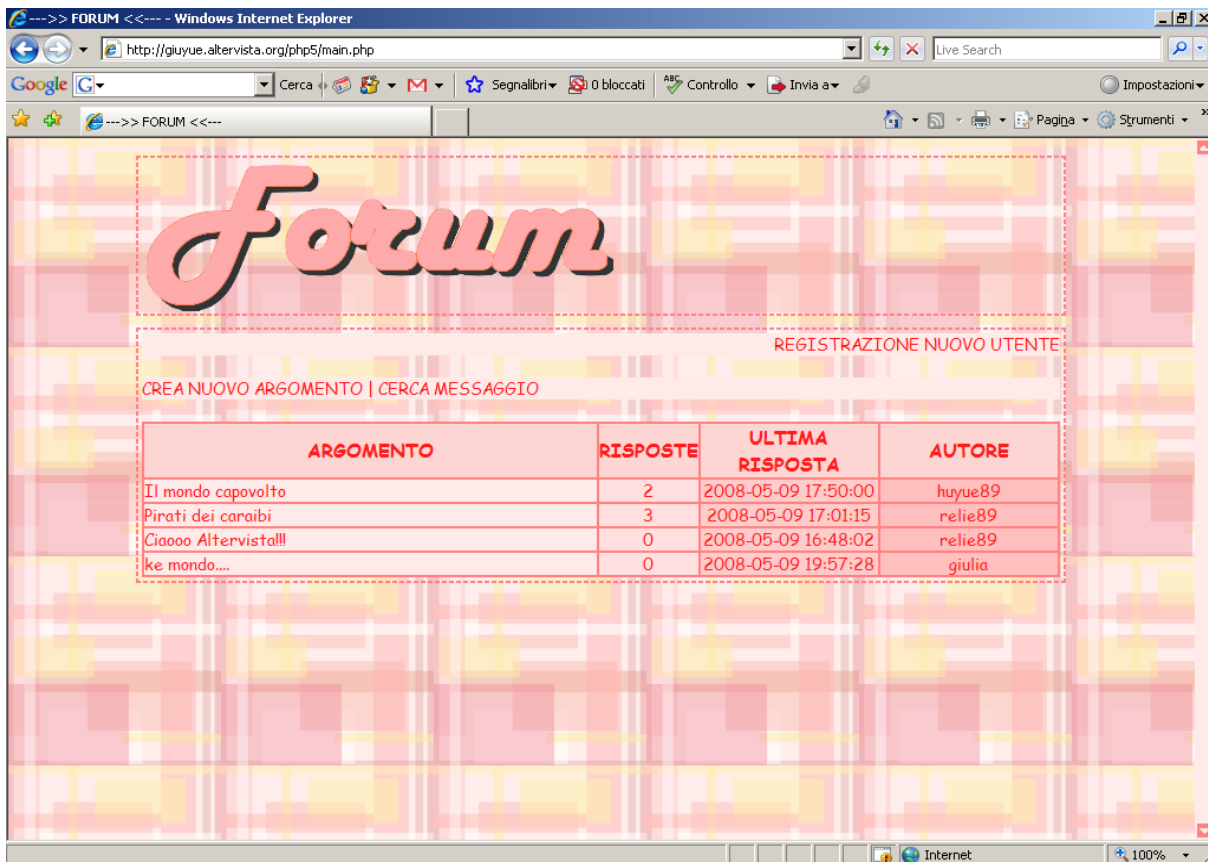
Utente(Nome, Password, RegistratoIl, Località, NumMessaggi)

Messaggio(ID, Autore, Argomento, Titolo, CorpoMessaggio, InviatoIl)

# Scelte implementative

## Descrizione dell'interfaccia:

Si tratta come già detto in precedenza di una pagina web che si presenta inizialmente all'utente nella seguente forma:



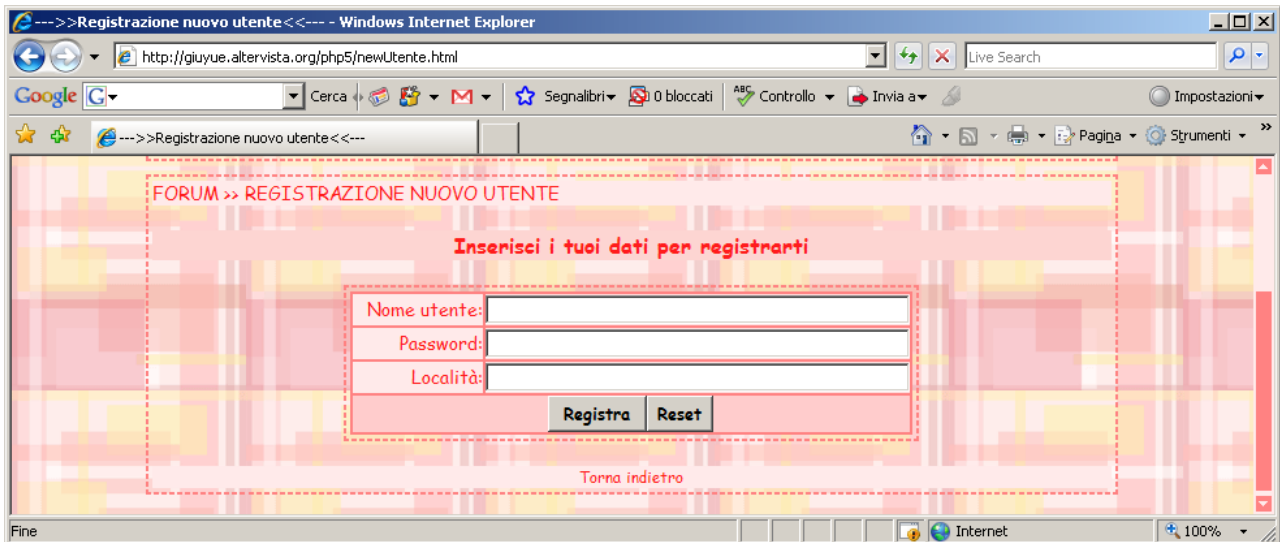
Come si può notare, inizialmente viene mostrato il contenuto del forum in modo schematico, ossia visualizzando in forma tabellare le informazioni relative agli argomenti.

In questa prima pagina viene data all'utente la possibilità di effettuare la registrazione al database, e di poter creare un nuovo argomento o cercare un messaggio.

O ancora più semplicemente è possibile visualizzare tutti i messaggi contenuti per un qualsiasi argomento scelto.

Per ciascuna delle operazioni descritte è sufficiente cliccare sopra la scritta (un semplice link) e verrà eseguita l'operazione richiesta.

Nel caso della registrazione di un utente, verrà presentata la seguente schermata:

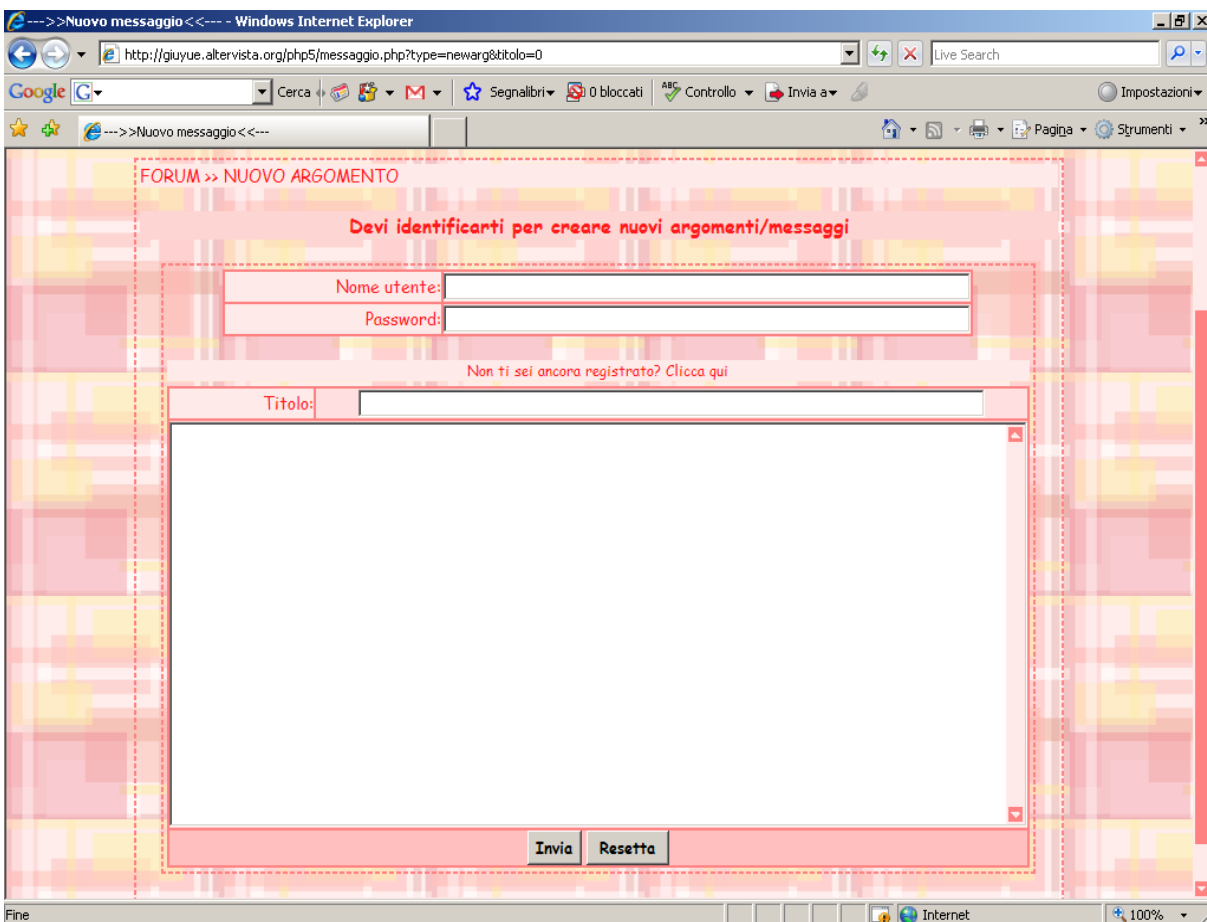


The screenshot shows a Windows Internet Explorer browser window with the address bar displaying `http://giuyue.altervista.org/php5/newUtente.html`. The page title is "Registrazione nuovo utente". The main content area has a red dashed border and contains the following elements:

- Header: "FORUM >> REGISTRAZIONE NUOVO UTENTE"
- Instruction: "Inserisci i tuoi dati per registrarti"
- Form fields:
  - "Nome utente:" with an input field
  - "Password:" with an input field
  - "Località:" with an input field
- Buttons: "Registra" and "Reset" (both in grey)
- Link: "Torna indietro" (in red)

In questo caso vengono forniti dei campi di testo per l'inserimento dati e due bottoni tramite i quali è possibile confermare la registrazione, che porterà ad una conseguente immissione dei dati nel database, oppure resettare i dati digitati.

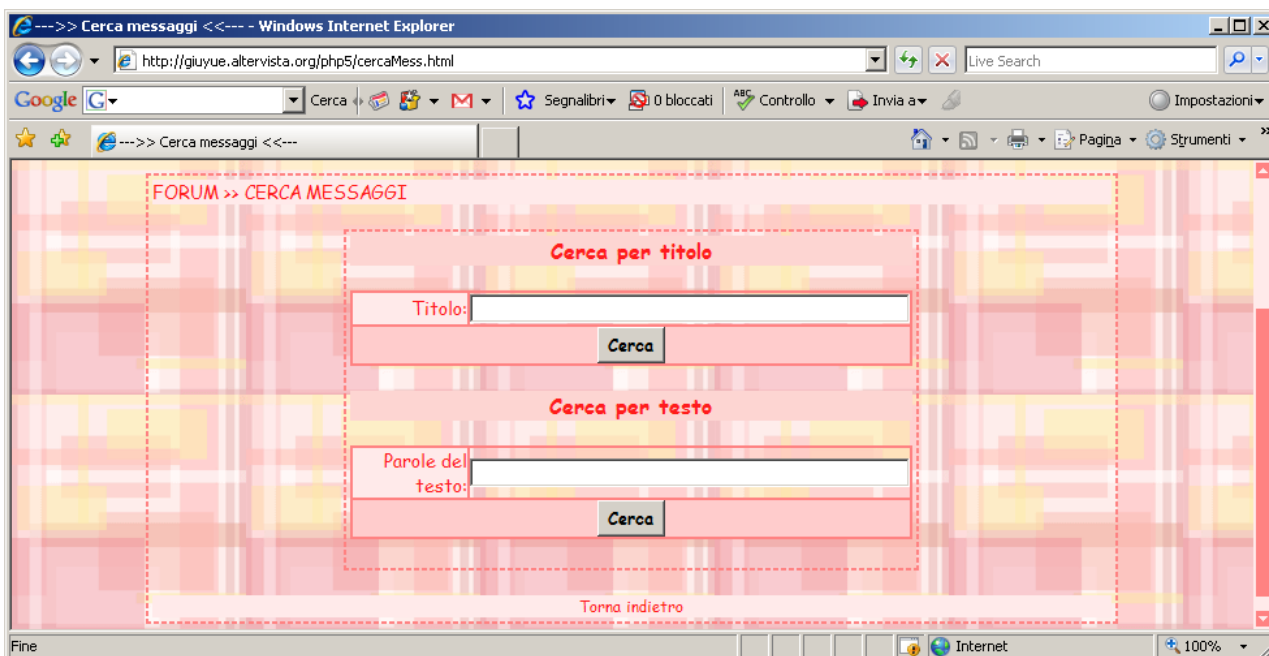
Nel caso in cui l'utente voglia, invece, creare un nuovo argomento, viene visualizzata la seguente schermata con la richiesta di alcuni dati, inseribili sempre tramite campi di testo, e con i due bottoni per confermare o meno l'operazione:



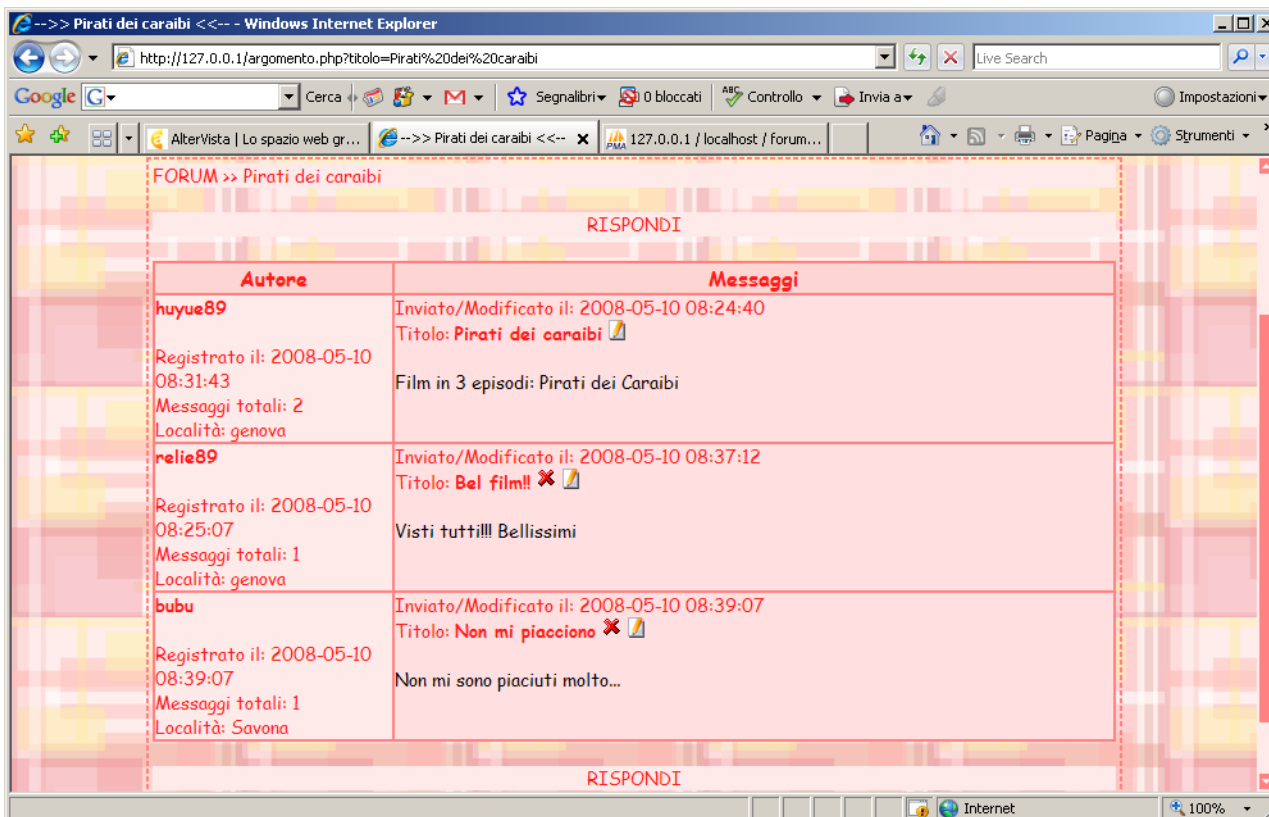
The screenshot shows a Windows Internet Explorer browser window with the address bar displaying `http://giuyue.altervista.org/php5/messaggio.php?type=newarg&titolo=0`. The page title is "Nuovo messaggio". The main content area has a red dashed border and contains the following elements:

- Header: "FORUM >> NUOVO ARGOMENTO"
- Instruction: "Devi identificarti per creare nuovi argomenti/messaggi"
- Form fields:
  - "Nome utente:" with an input field
  - "Password:" with an input field
- Text: "Non ti sei ancora registrato? Clicca qui" (in red)
- Form fields:
  - "Titolo:" with an input field
  - A large text area for the message content
- Buttons: "Invia" and "Resetta" (both in grey)

Per la ricerca di un messaggio invece viene visualizzata all'utente la seguente schermata:

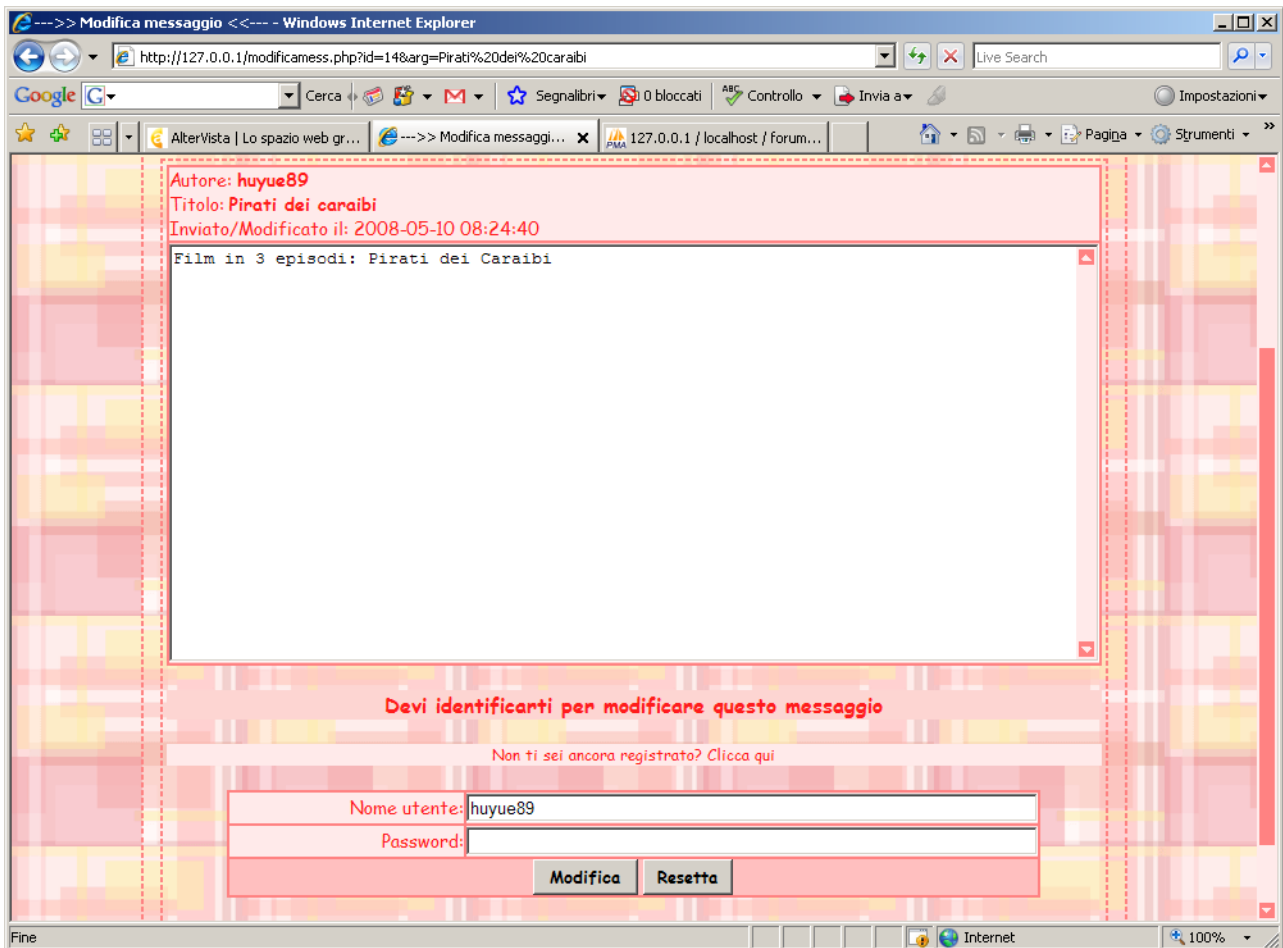


Invece, quando si clicca su di un argomento, viene visualizzato l'elenco dei post relativi ad esso con tutte le informazioni necessarie. Come ad esempio:



Come risulta intuitivo, i bottoni con icona a forma di 'x' consentono l'eliminazione, quelli con icona che illustra foglio e matita, consentono la modifica del contenuto del messaggio.

Ogni modifica avviene con la seguente visualizzazione di schermata:



Come visibile dall'immagine, viene riportato il messaggio, che era già stato scritto, in un'area di testo dove è possibile per l'utente apportare modifiche. Inoltre, come già illustrato, viene sempre richiesta l'immissione della password, mentre il nome non è necessario, perché solo l'autore del messaggio è in grado di cancellarlo.

Da ogni pagina è possibile tornare indietro, per poter riprendere l'attività precedente.

## Tecnologie e linguaggi utilizzati:

La realizzazione delle pagine per la gestione del forum è stata fatta tramite linguaggio di marcatura html per le pagine statiche (comuni "form" per l'invio di informazioni al server) e **php** per le pagine dove è necessario collegarsi al database; tale scelta è dovuta al fatto che php è un *prodotto open source e gratuito*, compatibile con quasi tutti i browser web e piattaforme diverse, con la possibilità di supportare una completa gamma di databases; grosse potenzialità, poi, sono la semplicità di apprendimento e l'elasticità delle procedure.

E' stata preferita, dunque, un'elaborazione che sposta la **logica dell'applicazione sul server**: in questo modo è possibile ridurre i requisiti che il browser utilizzato dall'utente deve soddisfare (si evita di imporre il supporto di applet Java o di altre tecnologie client-side). Si è contemporaneamente ritenuto non giustificato il ricorso ad un *Application Server* infatti, non dovendo realizzare un'applicazione per web multilivello ed enterprise, con alto grado di complessità ed altissima scalabilità, ci pareva eccessivo l'uso di *software* che fornisce infrastruttura e funzionalità di supporto, sviluppo ed esecuzione per applicazioni e componenti server in un contesto distribuito.

La scelta è dovuta, allora, al bisogno di un linguaggio che possa essere elaborato e interpretato dal

web server dove è stato memorizzato il database, in grado di restituire al browser codice html “creato al volo” senza richiedere nessuna elaborazione lato client: al browser viene richiesto esclusivamente di svolgere la propria funzione di visualizzazione di pagine web.

Nel confronto tra le tecnologie che supportano siti web dinamici e consentono l'implementazione di **pagine attive**, collocandosi dunque nel contesto delle *server pages* (ASP, JSP, PHP) caratterizzate dai seguenti elementi:

- Sono **dinamiche** in quanto il loro output può dipendere da fattori quali il momento in cui vengono eseguite, il risultato di operazioni sul database, informazioni fornite dall'utente e così via.
- Sono tecnologie **lato server**, nel senso che l'esecuzione delle componenti dinamiche avviene sul server e ciò che viene inviato al browser è esclusivamente il risultato prodotto dall'esecuzione della pagina.
- Consentono l'immersione del codice nell'HTML della pagina; in tal modo i contenuti dinamici della pagina vengono generati da frammenti di programma mentre le parti statiche, che vengono lasciate immutate, ne definiscono l'aspetto grafico.

si è scelto il linguaggio PHP anche con attenzione alla scalabilità: un modulo PHP dentro Apache scala molto meglio in quanto ottimizza la risposta alle richieste contemporanee ricevute (per esempio: mentre pagine *JSP continuerebbero ad aprire collegamenti con un data base*, il modulo PHP apre un certo numero di collegamenti e li riusa).

Il lato server quindi deve essere in grado di interpretare il **linguaggio php** permettendo agli utenti di effettuare le diverse operazioni interagendo con il database remoto e ottenendo come risposta pagine html.

In questo caso, per la realizzazione finale, tra i possibili fornitori di “web hosting free” si è scelto Altervista (<http://www.altervista.org>) che, oltre a fornire gratuitamente spazio web, permette gratuitamente la realizzazione e gestione di DB MySQL con interfaccia phpMyAdmin.

Per la fase di test, in locale, invece è stato utilizzato il server Apache di EasyPhp e phpMyAdmin.

## DataBase utilizzato

```
-- phpMyAdmin SQL Dump
-- version 2.9.1
-- http://www.phpmyadmin.net
--
-- Host: localhost
-- Generato il: 10 Mag, 2008 at 09:28 AM
-- Versione MySQL: 5.0.51
-- Versione PHP: 4.4.8
--
-- Database: `my_giuyue`
--
-----
```



```

--
-- Struttura della tabella `argomento`
--
CREATE TABLE `argomento` (
  `Titolo` varchar(50) NOT NULL default '',
  `Autore` varchar(20) NOT NULL default '',
  `NumRisposte` int(11) NOT NULL default '0',
  PRIMARY KEY (`Titolo`),
  KEY `Autore` (`Autore`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

--
-- Dump dei dati per la tabella `argomento`
--
INSERT INTO `argomento` (`Titolo`, `Autore`, `NumRisposte`) VALUES
('Il mondo capovolto', 'huyue89', 2),
('Pirati dei caraibi', 'relie89', 3),
('Ciaooo Altervista!!!', 'relie89', 0),
('ke mondo....', 'giulia', 0);

-----

--
-- Struttura della tabella `messaggio`
--

CREATE TABLE `messaggio` (
  `ID` int(11) NOT NULL auto_increment,
  `Autore` varchar(20) NOT NULL default '',
  `Argomento` varchar(50) NOT NULL default '',
  `Titolo` varchar(50) NOT NULL default '',
  `InviatoIl` timestamp NULL default NULL on update CURRENT_TIMESTAMP,
  `CorpoMessaggio` text NOT NULL,
  PRIMARY KEY (`ID`),
  KEY `Autore` (`Autore`),
  KEY `Argomento` (`Argomento`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;

--

```

```
-- Dump dei dati per la tabella `messaggio`
--
INSERT INTO `messaggio` (`ID`, `Autore`, `Argomento`, `Titolo`,
`InviatoIl`, `CorpoMessaggio`) VALUES
(1, 'huyue89', 'Il mondo capovolto', 'Il mondo capovolto', '2008-05-09
17:50:00', 'Questo mondo "" capovolto'),
(2, 'relie89', 'Il mondo capovolto', 'ok', '2008-04-19 09:34:23', 'prova
prova'),
(5, 'relie89', 'Pirati dei caraibi', 'Pirati dei caraibi', '2008-04-19
09:48:31', 'ciao'),
(6, 'huyue89', 'Pirati dei caraibi', 'pirati pirati!', '2008-04-19
09:52:10', 'arrivano i pirati!!! -o-'),
(9, 'relie89', 'Il mondo capovolto', 'ciao', '2008-04-19 10:31:09',
'ciaociao'),
(16, 'giulia', 'ke mondo....', 'ke mondo....', '2008-05-09 19:57:28',
'ciao a ttt...'),
(12, 'relie89', 'Ciao Altervista!!!', 'Ciao Altervista!!!', '2008-
05-09 16:48:02', 'Ciao Altervista!!!<br>Ciao Altervista!!!'),
(13, 'relie89', 'Pirati dei caraibi', 'ciao', '2008-05-09 16:55:21',
'ciao'),
(15, 'huyue89', 'Pirati dei caraibi', 'bubu', '2008-05-09 17:01:15',
'bubu');
```

```
-----
--
```

```
-- Struttura della tabella `utente`
```

```
--
CREATE TABLE `utente` (
  `Nome` varchar(20) NOT NULL default '',
  `Password` varchar(20) NOT NULL default '',
  `RegistratoIl` timestamp NULL default NULL on update CURRENT_TIMESTAMP,
  `Località` varchar(50) default NULL,
  `NumMessaggi` int(11) default NULL,
  PRIMARY KEY (`Nome`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
--
-- Dump dei dati per la tabella `utente`
```

```
--
INSERT INTO `utente` (`Nome`, `Password`, `RegistratoIl`, `Località`,
`NumMessaggi`) VALUES
('huyue89', '4321', '2008-05-09 17:01:15', 'genova', 3),
('relie89', '1234', '2008-05-09 17:06:07', 'genova', 5),
('giulia', 'gigi', '2008-05-09 20:01:06', 'genova', 1);
```

# SQL

## Query di creazione:

Query di selezione dei dati per visualizzazioni:

```
select * from nomeTabella
```

Query di selezione da più tabelle per visualizzazione di post, dove \$titolo è il titolo dell'argomento:

```
select * from messaggio join utente on messaggio.Autore=utente.Nome  
where Argomento = '$titolo' order by ID
```

Query per trovare l'ultimo messaggio aggiunto/modificato per ogni argomento:

```
select InviatoIl from messaggio, argomento where messaggio.Argomento =  
'$riga[Titolo]' order by InviatoIl DESC
```

Query di inserimento dei dati di registrazione di un utente:

```
insert into utente values('$nome', '$pass', NOW(), '$loco', 0)
```

Query di inserimento di nuovi messaggi:

Nuovi argomenti:

```
insert into argomento values('$titolo', '$nome', 0)
```

Nuovi messaggi:

```
insert into messaggio  
values(NULL, '$nome', '$titolo', '$titolo', NOW(), '$corpoTesto')
```

Ogni volta che viene inserito un messaggio, viene aggiornato il numero di risposte degli argomenti e numero di messaggi dell'utente:

```
update argomento set NumRisposte = $n where Titolo = '$arg'
```

```
update utente set NumMessaggi = $n where nome = '$nome'
```

Modifica di un messaggio:

```
update messaggio set CorpoMessaggio = '$corpoTesto' where ID = $id
```

Eliminazione di un messaggio:

```
delete from messaggio where ID = '$id'
```

## Query significative:

Poiché è stata prevista anche la funzione di ricerca, la query di ricerca (anche se molto restrittiva) è la seguente:

```
SELECT * FROM messaggio where Titolo LIKE '$chiave%' ORDER BY ID
```

Si consiglia all'amministratore di utilizzare la seguente query per controllare il funzionamento del conteggio dei messaggi di utenti e risposte di argomenti:

Messaggi utenti:

```
select Autore, COUNT(*) from messaggio, utente where messaggio.Autore =  
utente.Nome group by Autore
```

Risposte agli argomenti (il numero visualizzato nella pagina html è questo numero -1 perché non si conta il primo messaggio come risposta):

```
select Argomento, COUNT(*) from messaggio, argomento where  
messaggio.Argomento = argomento.Titolo group by Argomento
```

## Manuale d'uso

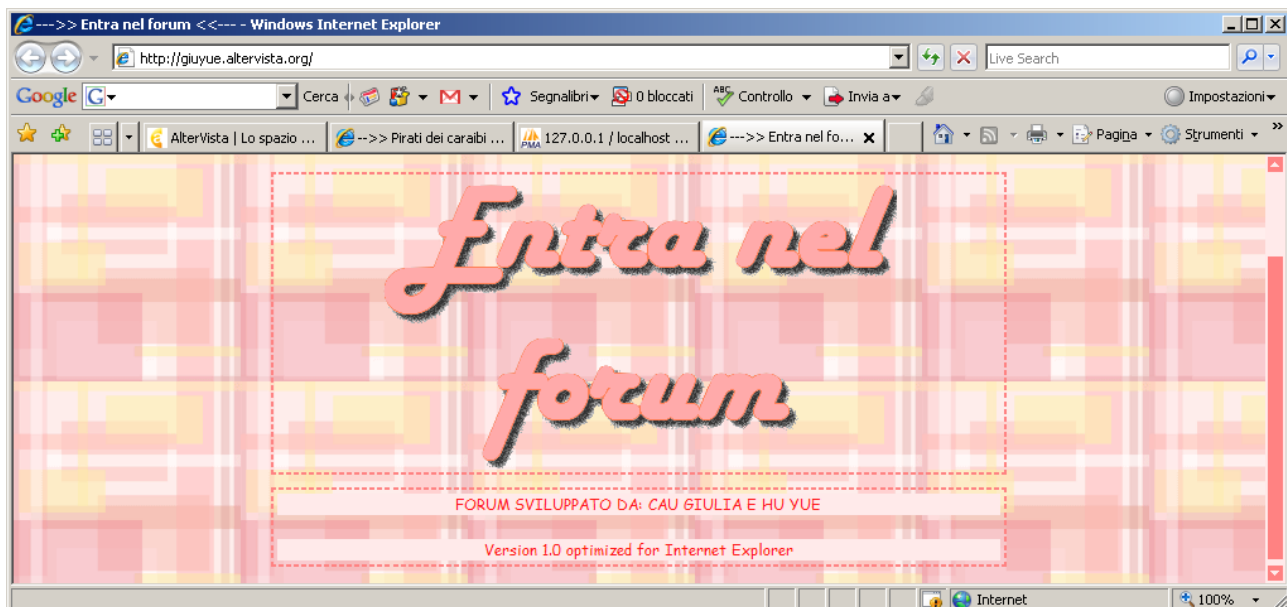
Nella descrizione dell'interfaccia è stata mostrata la realizzazione del progetto per la comunicazione con l'utente. Le schermate fornite, forniscono già di per sé esempi concreti per imparare gradualmente l'utilizzo del sistema.

Poiché, com'è già stato spiegato, il tutto è simile alla consultazione di un sito web in cui si trova la gestione del forum, la modalità di comunicazione con l'utente è del tutto semplice ed immediata.

Riguardo all'utilizzo dell'intero sistema, una volta realizzato e testato il DB in locale, trattandosi di un insieme di pagine ASCII, si è scelto di caricare tutte le pagine sul sito web di Altervista ed importare le diverse tabelle nel database (my\_ *nomeutente*) gratuitamente disponibile.

Perciò è sufficiente possedere una connessione ad internet, connettersi tramite un browser alla pagina: <http://giuyue.altervista.org>, e da qui in poi è assolutamente possibile poter usufruire della gestione del forum realizzata nel progetto.

Entrando dalla pagina principale si accede alla pagina dove viene visualizzato il forum:



Si ricordano nuovamente i seguenti punti fondamentali per l'utilizzo di questo forum:

- per leggere e cercare messaggi non è necessaria la registrazione
- per effettuare tutte le altre operazioni quali scrittura, modifica e cancellazione di messaggi è necessario registrarsi con un nome utente e una password
- per creare un nuovo argomento basta scrivere un nuovo messaggio che rappresenta appunto il primo messaggio della discussione, il titolo di questo messaggio è il titolo dell'argomento
- non è permessa la cancellazione del primo messaggio di un argomento, tale operazione è permessa solo all'amministratore
- non è possibile eliminare un utente creato, nemmeno dall'utente stesso, eventualmente tale richiesta deve essere rivolta all'amministratore
- è necessario identificarsi ogni volta che si intende effettuare un'operazione, non esiste quindi, la funzione di login permanente e/o temporaneo
- la ricerca di messaggi può avvenire tramite titolo o testo del messaggio, tuttavia è una ricerca molto restrittiva, da migliorare eventualmente successivamente

## Manuale per la fase di testing:

Per effettuare il test del funzionamento del forum è necessario dotarsi ad esempio di ambiente EasyPhp (Apache, MySQL, phpMyAdmin, per Windows). Una volta disponibile, creare il database 'forum' di test con nessuna password ed utente qualsiasi:

```
CREATE USER ''@ 'localhost';

GRANT ALL PRIVILEGES ON * . * TO ''@ 'localhost' WITH GRANT OPTION
MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0 MAX_UPDATES_PER_HOUR 0
MAX_USER_CONNECTIONS 0 ;
```

importare il dump fornito per creare le tabelle del database MySQL, e copiare tutto il materiale fornito (pagine html, pagine php e immagini) nella cartella www di EasyPhp.

Quindi accedere al sito locale di EasyPhp: dovrebbe visualizzarsi la pagina principale; in caso contrario, se si è copiato il materiale in una sottocartella (ad esempio di nome *forum*), selezionare tale sottocartella e scegliere dalla lista dei file che si presenta nella pagina, la pagina principale index.html oppure main.php.

Nb: mentre per il test in locale si è fatto uso di versione aggiornata (EasyPhp 2.0b1) creando tabelle InnoDB ed impostando chiavi esterne con gestione dell'integrità referenziale, per la distribuzione del prodotto (come relazionato) si è privilegiata la retrocompatibilità usando tabelle MyISAM. Ad esempio, con web account free su Lycos (<http://www.tripod.lycos.com/>), si dispone gratuitamente di database con nome *nomeuser\_db\_it* ma versione di phpMyAdmin non aggiornata.