

**ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA
CAMPUS DI CESENA
SCUOLA DI SCIENZE**

CORSO DI LAUREA IN SCIENZE DELL'INFORMAZIONE

**SISTEMA DI VOTAZIONE
CON GESTIONE DEGLI UTENTI IN UN SITO WEB**

**Relazione finale in
MOBILE WEB DESIGN**

Relatore

Dott. MIRKO RAVAIOLI

Presentata da

CLAUDIA MARIANI

Sessione III

Anno Accademico 2012-2013

*A te NONNA...
Esempio di vita,
Donna di amore,
forza e rispetto...*

INDICE

INTRODUZIONE	1
 CAPITOLO 1	
I sistemi CMS e Joomla	3
1.1 Che cos'è il CMS	3
1.2 Che cos'è Joomla	4
1.3 Linguaggio utilizzato da Joomla	7
1.3.1 <i>Linguaggio PHP</i>	7
1.3.2 <i>MySQL</i>	9
 CAPITOLO 2	
Sistema di votazione sviluppato con Plugin	11
2.1 Il sistema di votazione	11
2.1.1 <i>Errori e Problematiche legate al sistema di votazione</i>	12
2.2 Che cosa è un Plugin	14
2.2.1 <i>Diverse tipologie di Plugin</i>	15
2.2.2 <i>Eventi di un Plugin Content</i>	18
 CAPITOLO 3	
Creazione di Plugin Content in Joomla	19
3.1 Creazione dei file necessari per il Plugin	19
3.1.1 <i>Creazione del file XML</i>	19
3.1.2 <i>Creazione del file PHP</i>	24
3.1.3 <i>Creazione del file index.html</i>	25
3.2 Aggiunta di ulteriori file necessari per lo sviluppo del Plugin	26
3.2.1 <i>File CSS</i>	26
3.2.2 <i>File Javascript js</i>	28
3.2.3 <i>File ajax.php</i>	29
3.3 Gestione del database in Joomla.....	30
3.3.1 <i>Creazione Tabelle</i>	30
3.3.2 <i>Inserimento record</i>	32
3.3.3 <i>Ricavare i dati inseriti nella tabella</i>	33

3.3.4	<i>Modifica valori</i>	33	
3.4	Funzione di accesso al database in Joomla	34	
CAPITOLO 4			
Gestione degli utenti			35
4.1	Tecniche di sicurezza per la votazione	35	
4.1.1	<i>La tecnica dei cookies</i>	35	
4.1.2	<i>La tecnica del controllo dell'IP(Internet Protocol Address)</i>	36	
4.1.3	<i>La tecnica di registrazione dell'utente</i>	37	
4.2	Gestione degli utenti in Joomla	37	
4.3	Programmazione del Modulo Login in Joomla	42	
CAPITOLO 5			
Installazione e funzionamento del Plugin			47
5.1	Installazione Plugin	47	
5.2	Abilitare e testare il Plugin	48	
CONCLUSIONI			51
BIBLIOGRAFIA			52

INTRODUZIONE

Lo scopo di questa tesi è di permettere ad un utente, che visita un qualunque sito web, di effettuare una valutazione o meglio, di attribuire un indice di gradimento, al termine della lettura di un articolo, di un video o di un'immagine.

È stata effettuata un'attenta e accurata valutazione dei sistemi che potevano essere presi in esame tramite materiale trovato su internet, cartaceo fornito dal professore che mi ha supportato nella creazione della tesi e ricerche bibliotecarie.

Il passo successivo è stato quello della scelta del sito da utilizzare come "cavia" per il mio progetto di valutazione.

Sono stati presi in esame vari siti internet e valutati sotto ogni aspetto: la presenza di una costante pubblicazione di articoli da poter votare, la presenza di numerose immagini da visualizzare, la possibilità da parte dell'utente di poter interagire in modo attivo e l'aggiornamento continuo del sito.

Tutto questo perché maggiore è la visualizzazione da parte dell'utente, maggiore è la sua partecipazione e maggiore diventerà la facilità di trovare il sito tramite motore di ricerca.

La scelta è ricaduta sul sito della società sportiva USD San Zaccaria (www.usdsanzaccaria.it) per due semplici motivi: il primo molto pratico, è che per lavoro, io stessa, in collaborazione con un mio collega, stavo lavorando alla progettazione di questo sito su richiesta della società. E il secondo è un motivo strettamente personale dal momento che gioco a calcio per la società USD San Zaccaria che milita nel campionato Nazionale di serie B.

Una volta scelto il sito da utilizzare sono passata ad un'analisi più specifica del sito in questione.

Il Content Management System (CMS) utilizzato nel sito dell' USD San Zaccaria è JOOMLA quindi è stato necessario scegliere un "Plugin" che fosse adatto a questo software.

Nell'arco della tesi saranno poi sviluppati i capitoli in cui verranno articolate le spiegazioni relative alla descrizione di Joomla, alla spiegazione di che cosa si intende per software di Content Management System o CMS e alla programmazione di un Plugin.

È stato poi progettato un sistema di valutazione per gli articoli che si trovano sul

sito da me selezionato e utilizzato. Si è scelto di ridurre il campo di applicazione del sistema di valutazione da parte degli utenti, alla sezione riguardante le news del campionato femminile di serie B, per poter avere una più rapida e facile visualizzazione della quantità di utenti utilizzatori del sistema.

È stato poi creato, all'interno del sito, una parte chiamata "login degli utenti" in cui il visitatore deve effettuare una registrazione, inserendo i dati personali e creando un proprio profilo, compreso di username e password. Tutto ciò è stato creato affinché la votazione dell'articolo sia realistica, in quanto con questo sistema l'utente vota una sola volta.

Questa tesi è stata progettata con un occhio rivolto al futuro, e più precisamente al mio futuro; perché nonostante il campo dell'informatica sia molto ampio, credo che quello più interessante e affine alle mie capacità sia la realizzazione dei siti web.

La realizzazione di questa tesi mi ha permesso inoltre di conoscere, imparare e applicare nuovi tipi di linguaggi di programmazione.

CAPITOLO 1

I SISTEMI CMS E JOOMLA

1.1 Che cos'è il CMS

Il CMS o Content Management System è un sistema di gestione dei contenuti o meglio, è uno strumento software, installato su un server web, il cui compito è facilitare la gestione dei contenuti dei siti web.

Il CMS viene utilizzato dai webmaster per gestire un sito web e per poter così evitare di programmare dinamicamente l'intero software lato server che gestisce il sito stesso.

Parlando in termini più tecnici, un CMS è un'applicazione lato server, che si appoggia su un database già esistente utilizzato per lo stoccaggio dei contenuti.

L'applicazione di tale sistema è suddivisa in due sezioni: una amministrativa e una applicativa.

La prima, chiamata anche "back end", serve ad organizzare e a supervisionare la produzione dei contenuti; la seconda, o "front end", viene usata dall'utente per fruire dei contenuti e delle applicazioni che sono all'interno del sito web.

L'amministratore del CMS gestisce, dal proprio terminale, i contenuti che sono da inserire o da modificare attraverso l'utilizzo di un pannello di interfaccia e controllo.

Questo programma consente la pubblicazione, l'editing e la modifica dei contenuti, nonché la manutenzione da un'interfaccia centrale.

Generalmente, offre un'interfaccia di amministrazione con cui l'utente può gestire ogni aspetto del sito web, senza necessariamente essere in possesso di conoscenze tecniche particolari riferite alla programmazione Web. Tali sistemi di gestione dei contenuti forniscono le procedure per gestire il flusso di lavoro in un ambiente collaborativo.

I CMS sono spesso utilizzati per i siti web che contengono al loro interno blog e notizie, oppure, siti utilizzati per lo shopping; ma allo stesso tempo anche molti siti web aziendali e di marketing fanno uso dei CMS.

Questi possono essere semplicemente generici, che tendono ad essere più flessibili, in modo da poter consentire la pubblicazione di diversi tipi di contenuti; oppure possiamo avere dei CMS specializzati: cioè appositamente progettati per un determinato tipo di contenuti come potrebbe essere ad esempio un'enciclopedia on-

line, un particolare forum o blog oppure una rivista.

I CMS possono essere realizzati tramite programmazione e utilizzo di vari linguaggi web. Il tipo che viene scelto e utilizzato non provoca differenze a livello di funzionalità; quelli più comunemente usati sono ASP, PHP, Microsoft .NET.

1.2 Che cos'è Joomla

Possiamo avere numerose tipologie di CMS ma come in tutti gli ambiti ci sono quelli più selezionati: Joomla (vedi figura 1.1), insieme a Drupal e Wordpress, è il Web Content Management System open source più usato nel mondo.



Figura 1.1 - Logo Joomla

Le caratteristiche di Joomla hanno fatto sì che venisse scelto tra gli altri per la progettazione del sito web relativo a questa tesi.

La parola Joomla deriva dalla parola Jumla in lingua africana Swahili e significa “tutti insieme”.

Joomla è un pluripremiato sistema di gestione dei contenuti, che consente di creare siti web e potenti applicazioni on-line. Molti aspetti, compresa la sua facilità d’uso ed estendibilità, hanno reso Joomla il più popolare software.

È un sistema gratuito rilasciato sotto licenza Open Source, che permette di creare facilmente e velocemente siti Web, blog, portali Internet e community on-line senza dover mettere mano ad una sola riga di codice.

Come tanti altri progetti open source, è in continua evoluzione. È imprevedibile, qualche volta indescrivibile, in parte controverso. Nonostante ciò, ha riscosso un enorme successo e da cinque anni è divenuto talmente popolare da essere usato da milioni di utenti nel mondo.

Joomla è usato in tutto il mondo per semplici homepage, per social web e per siti più complessi. È facile da installare, da gestire e molto affidabile.

Joomla è un’applicazione relativamente “giovane”, la sua prima versione risale

infatti al 2005; ma la sua struttura si basa su un altro CMS, Mambo, disponibile in rete già dal 2000.

Joomla è quindi di fatto un “fork”, cioè un progetto nato da un'altra piattaforma a causa di una scissione tra gli sviluppatori. Molti dei programmatori che lavoravano su Mambo si sono, infatti, allontanati dal progetto per aderire all'organizzazione no profit Open Source Matters che fa capo a Joomla.

Grazie alla licenza GNU GPL versione 2, Joomla può essere scaricato, utilizzato e modificato liberamente. Chiunque, infatti, nel rispetto della suddetta licenza, può usufruire del prodotto e agire sul suo codice in modo da adattarlo alle proprie esigenze; inoltre, si tratta non soltanto di un prodotto libero ma anche di un'applicazione gratuita per il cui utilizzo non è previsto alcun esborso di denaro.

Joomla è un prodotto Open Source molto utilizzato soprattutto perché completo e semplice da installare, e offre una procedura guidata per la configurazione del sito web.

Un punto di forza di questo CMS è il potente pannello di amministrazione (vedi figura 1.2), semplice, ma ricco di funzioni; Joomla, infatti, separa in modo netto l'aspetto grafico del CMS dai contenuti e dagli strumenti di amministrazione.

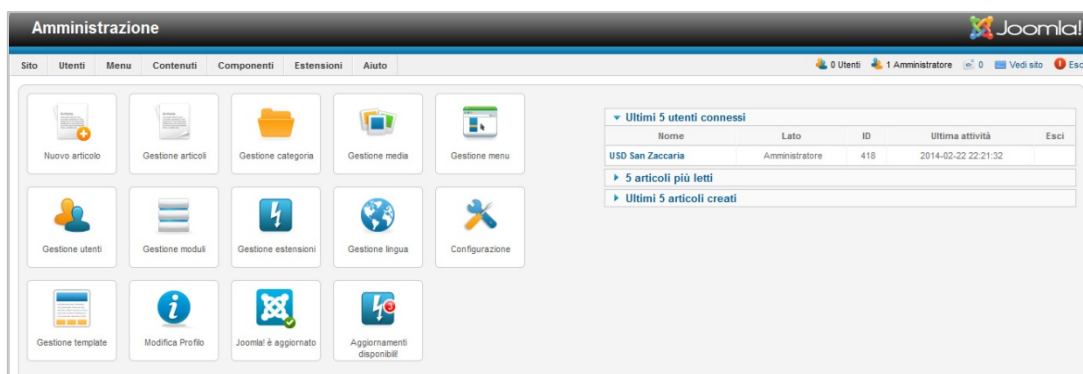


Figura 1.2 - Pannello amministrativo di Joomla

Tutti siti gestiti da CMS sono dotati di pannello di amministrazione attraverso il quale il webmaster o il cliente finale possono tenere aggiornati i contenuti e inserirne di nuovi, senza dover mai accedere al codice sorgente del sito, e quindi senza particolari conoscenze di programmazione web.

Per poter utilizzare Joomla, il primo passo è scaricarlo; effettuato ciò si otterrà un file zip di circa sette megabyte che deve essere installato su un server web (ad

esempio /htdocs). Eseguendo l'installazione vengono estratti tutti i file e viene inserito il contenuto di riempimento nel database.

Per installare Joomla si ha bisogno di un ambiente di lavoro chiamato WAMP o LAMP. Questi sono due sistemi client-server analoghi; il primo viene utilizzato per identificare l'ambiente Windows, da qui il nome Windows Apache (nome server) MySQL PHP, mentre il secondo serve per identificare l'ambiente Linux, dal quale prende il nome.

Un sito web Joomla è composto da vari elementi che cooperano per generare le pagine web. I tre elementi principali sono: il contenuto, i template e i moduli. Il contenuto è l'aspetto centrale del sito web, il template controlla la presentazione del contenuto e i moduli aggiungono funzionalità dinamiche intorno al contenuto principale della pagina.

La parte più importante di un sito web è il contenuto, ovvero, ciò che i visitatori guardano, e Joomla, in quanto CMS, aiuta a crearlo, pubblicarlo e gestirlo in modo efficiente. Il contenuto è organizzato in porzioni facili da gestire chiamate articoli. Il contenuto del corpo principale è generato da ciò che Joomla chiama componente. Il componente più grande e importante in Joomla è quello che gestisce tutti gli articoli ovvero le singole unità di contenuto del sito, e in effetti è così importante che spesso gli articoli stessi sono indicati con il termine di articoli del contenuto.

Un template è invece un insieme di regole sulla presentazione dei componenti e dei moduli in una pagina e il loro posizionamento sullo schermo. Il template determina il layout o posizionamento della pagina web e, insieme ai suoi file, determina anche, per esempio, il numero di colonne da utilizzare e il colore da assegnare ai titoli. Un template opera come un filtro che controlla gli aspetti di presentazione della pagina web, è privo di contenuto ma può includere loghi.

I moduli sono piccoli blocchi funzionali mostrati di solito intorno alla pagina principale della pagina, come un'intervista o sondaggio.

I moduli possono visualizzare altro contenuto del database che può essere o meno correlato al contenuto del corpo principale, implementare funzioni del sito come gestire lo stato di accesso, fornire la navigazione ad altre pagine o fornire una capacità di ricerca.

Una delle caratteristiche più entusiasmanti di Joomla è che le sue funzionalità possono essere estese all'infinito attraverso l'installazione di plugin e componenti

aggiuntivi.

1.3 Linguaggio utilizzato da Joomla

Ogni CMS si sviluppa attraverso un tipo di linguaggio: Joomla è realizzato tramite l'utilizzo del linguaggio in PHP, con il supporto del database server MySQL per l'archiviazione dei dati.

La combinazione di PHP e MySQL consente di creare qualsiasi tipo di sito web, rendendo la realizzazione più semplice e veloce.

1.3.1 Linguaggio PHP

PHP è nato da una necessità per le persone a sviluppare e mantenere siti web dinamici contenenti funzionalità client-server. Nel 1994, Rasmus Lerdorf ha creato una collezione di script open source Perl per uso personale, che sono stati riscritti in linguaggio C e poi trasformati in quello che PHP è oggi. Nel 1998, PHP è stato rilasciato nella sua terza versione, trasformandolo in uno strumento di sviluppo web in grado di competere con altri prodotti simili, come ad esempio le pagine di Microsoft Active Server (ASP) e Sun Java Server Pages (JSP).

PHP è un linguaggio di Scripting General-Purpose open source e specialmente indicato per lo sviluppo web.

PHP è un cosiddetto acronimo "ricorsivo", perché significa "PHP Hypertext Preprocessor" ("PHP Preprocessore di Iper testi"): è in breve un acronimo in cui una delle lettere sta per la sigla, cioè un acronimo che contiene se stesso.

"Preprocessore" è invece un termine informatico che si riferisce a un programma che compie alcune operazioni preliminari, prima dell'esecuzione del codice: il nostro linguaggio contiene il termine "preprocessore" perché prima di inviare la pagina HTML finale all'utente vengono interpretate, appunto, le istruzioni del codice. Quindi prima che la pagina HTML venga consegnata al nostro browser, il preprocessore PHP compie alcune importanti operazioni.

Ciò che distingue PHP da altri linguaggi di scripting del tipo client-side JavaScript, è che il codice viene eseguito nel server, generando HTML che sarà dopo inviato al client. Il client dovrebbe ricevere i risultati dell'esecuzione dello script, ma non potrà conoscere qual è il codice eseguito.

In parole povere significa che uno script JavaScript verrà interpretato dal browser,

che eseguirà il codice e si comporterà di conseguenza. Invece le operazioni di un linguaggio lato server come PHP saranno compiute dal server, che consegnerà al browser la pagina HTML, codificata dopo aver eseguito una serie di operazioni.

Si potrebbe persino configurare il web server per processare tutte i file HTML con PHP ed allora non ci sarebbe realmente alcun modo per gli utenti di sapere cosa contiene quel determinato server. Si può infatti notare che su una pagina di un sito web qualunque non c'è neppure una riga di codice PHP, ma soltanto HTML. Quindi non è possibile copiare il codice PHP da un altro sito.

A differenza di HTML, in cui il browser web utilizza tag e markup per generare una pagina, il codice PHP viene eseguito tra la pagina richiesta e il server web, aggiungendo e modificando l'output HTML di base. PHP rende lo sviluppo web facile, perché tutto il codice necessario è contenuto all'interno del framework PHP.

PHP è un linguaggio interpretato, piuttosto che compilato: i linguaggi compilati, prevedono che il codice sorgente venga passato per un compilatore che lo trasforma in una serie di procedure comprensibili al processore, noto come file eseguibile (con estensione .exe); mentre il linguaggio interpretato, non richiede alcuna operazione di compilazione prima di essere eseguito, cioè non genera un eseguibile, ma il codice sorgente viene eseguito direttamente dall'interprete ed è l'interprete a dialogare con il processore.

PHP è onnipresente e compatibile con tutti i principali sistemi operativi.

La vera bellezza del linguaggio PHP sta nella sua semplicità, essendo molto facile da conoscere e applicare, il che lo rende uno strumento ideale per i principianti di programmazione web.

I creatori di PHP hanno sviluppato un'infrastruttura che permette agli esperti programmatori C di estendere le capacità PHP. Come risultato, PHP si integra ora con tecnologie avanzate come XML, XSL, e componenti di Microsoft Technologies Object Model (COM).

PHP può girare praticamente su qualsiasi server Web, su qualsiasi sistema operativo (Windows o Unix/Linux, ma anche Mac e altri), e consente di interagire praticamente con qualsiasi tipo di database (SQLite, MySQL, PostgreSQL, SQL Server, Oracle, SyBase, Access e altri). Si può utilizzare per i più svariati tipi di progetti, dalla semplice home page dinamica fino al grande portale o al sito di e-commerce.

I requisiti per il suo utilizzo sono semplicemente: un ambiente di produzione basato su PHP, un DBMS MySQL e un Web Server.

Anche se PHP è ottimo per lo sviluppo di applicazioni web, non memorizza le informazioni di per sé. Per questo, avete bisogno di un database. Il database di scelta per gli sviluppatori PHP è MySQL.

1.3.2 MySQL

MySQL è un DBMS (Database Management System), ovvero un sistema di gestione di basi di dati, fondato sul modello relazionale.

Un DBMS può contenere e gestire più database gestendo simultaneamente le richieste di diversi eventi.

I database o base di dati sono delle strutture nelle quali è possibile memorizzare grandi quantità di informazioni, per poi ricavarle attraverso linguaggi di scripting come il PHP.

Il punto di forza di un database sta nella velocità con cui le informazioni vengono trovate; dato il loro largo utilizzo ne esistono diversi tipi a seconda dell'uso che ne dobbiamo fare.

Attualmente il più diffuso è sicuramente il modello relazionale, che prevede che le informazioni contenute nella nostra base di dati siano rappresentate come una serie di relazioni tra diversi dati, e in particolare tra attributi e valori. Il modo migliore per rappresentare questa relazione sono le tabelle.

Queste svolgono una funzione di archivio al cui interno sono memorizzati i dati raccolti; ognuna di queste è suddivisa a sua volta in diversi "campi".

Le parti che formano le tabelle sono due: le colonne e le righe; le prime rappresentano le caratteristiche dell'elemento che si vuole memorizzare, e per ogni elemento inserito nelle tabelle viene creata una nuova riga.

MySQL è un software open source quindi è del tutto gratuito ed è supportato da Windows, da Linux e da Mac; ed è attualmente il DBMS più diffuso al mondo.

MySQL è un DBMS che sfrutta la sintassi del linguaggio SQL, "Structured Query Language", che significa "Linguaggio Strutturato di Interrogazioni".

SQL è il linguaggio più utilizzato per creare, recuperare, aggiornare e cancellare i dati dai sistemi di gestione di basi di dati.

Il programma che utilizzeremo per amministrare e gestire il nostro database sarà PhpMyAdmin; non è altro che un'interfaccia grafica che permette di visualizzare il contenuto del nostro database; inoltre supporta ulteriori funzioni di MySql come creare, modificare, cancellare intere tabelle, i campi e gli indici; fare un backup dei dati contenuti e visualizzare informazioni interessanti sul database.

CAPITOLO 2

SISTEMA DI VOTAZIONE SVILUPPATO CON PLUGIN

2.1 Il sistema di votazione

Un sistema di votazione è un sistema di statistiche di uso comune anche nella vita di tutti i giorni, con cui si cerca di far luce sui gusti delle persone sulla base di un campione di votanti. Anche per gli argomenti di natura informatica, esistono strumenti, o meglio, idee tali da permettere l'implementazione anche in formato elettronico.

Il sistema di votazione permette di tenere il sito vivace, attraverso il coinvolgimento del pubblico, e, permette all'utente di esprimere il proprio giudizio in merito.

Dare la possibilità agli utenti di interagire con un sito web è uno degli elementi chiave che possono portare al successo.

Lo scopo dell'inserimento del sistema di votazione è proprio quello di mantenere vivo il proprio sito web arricchendolo anche di contenuti.

Il sistema di votazione utilizzato all'interno del sito www.usdsanzaccaria.it è formato da una valutazione che va da un minimo di 1 ad un massimo di 5, con il vincolo di votare una volta sola.

In risposta a questa operazione avremo il totale dei voti ricevuti e il numero dei votanti.

Per poter votare un articolo è stato scelto un simpatico simbolismo, affine allo sport praticato a San Zaccaria: un grafico di palloni (vedi figura 2.1).



Figura 2.1 – Grafica di palloni sul sito web.

Per raccogliere ed elaborare i dati della votazione è stata utilizzata una tecnica che prevede la registrazione preventiva dell'utente per poter accedere alla votazione.

Ecco nei dettagli cosa prevede questa procedura:

compilando il modulo di login, dopo la registrazione dei propri dati, inserendo quindi, username e password, si ha una memorizzazione all'interno di una tabella, chiamata controllo, del numero identificativo dell'utente, chiamato user_id.

Dopo essersi registrato l'utente può, quindi, effettuare la votazione; ogni articolo ha un record nella tabella votazione che memorizza il totale dei voti ed il numero di votanti, quindi ogni volta che l'utente autenticato effettua la votazione, il suo voto viene aggiunto alla somma dei precedenti. Per fare in modo che l'utente non voti più di una volta, nella tabella controllo, vengono memorizzati i dati creando una terna unica, composta da un id (numero casuale crescente), id_articolo e user_id.

Ovviamente nemmeno in questo caso c'è certezza assoluta che qualcuno non utilizzi diversi account o dati anagrafici non corretti per ripetere più volte il voto, ma si tratta di un'operazione più lunga ed è possibile pensare che manchino le motivazioni perché un numero significativo di persone vi dedichi il proprio tempo.

2.1.1 Errori e problematiche legate al sistema di votazione

Uno degli errori più comuni è quello di modificare direttamente il codice sorgente di Joomla per aggiungere le informazioni desiderate. Questo approccio è errato, in quanto al prossimo aggiornamento di Joomla tutte le modifiche effettuate saranno sovrascritte.

Un altro metodo sbagliato di effettuare questa operazione è quello di modificare il codice del plugin User – profile; si tratta sempre di un plugin, ma è pur sempre una parte di Joomla e si ricadrebbe così nel problema precedente.

Questo plugin è concepito come un esempio per creare un plugin personalizzato, oppure è possibile utilizzarlo senza modifiche, se i campi di cui si ha la necessità sono già nella lista gestita dal plugin.

All'interno del web si possono trovare alcuni tutorial che consigliano di modificare la tabella degli utenti di Joomla aggiungendo nuovi campi; questo approccio era comunemente utilizzato fino a Joomla 1.5, ma, nelle nuove versioni, non è più considerato il metodo corretto di procedere.

La scelta migliore è quindi quella del Plugin essendo facilmente installabile e configurabile anche per un non-programmatore, e, se ben fatto, può essere compatibile con più versioni di Joomla.

Plugin è anche facilmente disinstallabile, senza lasciare residui o tabelle compresse in Joomla; e questi sono tutti vantaggi, soprattutto per chi non sa programmare ma si trova a dover gestire il sito.

Come già spiegato in precedenza, per plugin si intende una parte di codice che

viene eseguita in un determinato momento del caricamento della pagina, o, ad una determinata azione dell'utente.

La possibilità di eseguire il codice in un determinato momento e a seconda del verificarsi o meno di certe condizioni, ci dà la possibilità di estendere le funzionalità di Joomla, aggiungendo caratteristiche e operazioni che ci aiutano a risolvere determinati problemi: come ad esempio migliorare l'utilizzo del sito, aggiungendo delle informazioni ai contenuti, ma anche operazioni più complesse come migliorare le performance, effettuare il caching o un backup del database.

Gli unici limiti a questa estensibilità sono la nostra fantasia e la nostra abilità nella programmazione tramite PHP e la nostra conoscenza di Joomla.

Possiamo trovare i plugin installati di default in Joomla andando in "Estensioni" – "Gestione Plugin" nel backend del nostro sito Joomla.

Un plugin è uno strumento che può essere molto potente ed utile per il nostro sito Joomla.

Proprio per questo motivo si deve essere cauti nel realizzarlo, perché un plugin che viene progettato non tenendo a mente le performance o, peggio ancora, bypassando il framework di Joomla può essere dannoso e anche rischioso per un sito Joomla.

I due possibili problemi che possono verificarsi sono: perdita di performance e problemi di sicurezza e compatibilità.

Se un plugin effettua molte query in modo inefficiente o esegue cicli molto lunghi, utilizzando funzioni di PHP che consumano molte risorse, il risultato potrebbe rallentare sensibilmente il caricamento del sito.

Inoltre un Plugin che contiene codice non conforme al framework di Joomla rischia di compromettere la sicurezza del sito: se un programmatore con poca esperienza dovesse recuperare il contenuto di una variabile nella query string con PHP e quindi volesse visualizzarla nella pagina probabilmente utilizzerebbe un codice del genere:

```
<?php $variabile = $_GET["variabile"];?>
<p>La variabile nella query string &grave;
<?=$variabile?></p>
```

Questo codice contiene una falla di sicurezza e un possibile problema di compatibilità infatti una variabile proveniente dall'esterno va sempre controllata e filtrata prima di essere eseguita o visualizzata e non farlo significherebbe aprire delle falle.

Il framework di Joomla fornisce delle classi, metodi e funzioni che risolvono tutti i problemi di routine della programmazione di applicazioni in PHP.

Questo è il codice scritto precedentemente e in Joomla così da evitare problemi di sicurezza e compatibilità:

```
<?php $variabile = JRequest::getString("variabile", NULL);?>
<p>La variabile nella query string &egrave;
<?php echo $variabile?>
```

Questo codice fa uso del metodo statico `getString` della classe `JRequest` del framework di Joomla, che permette di recuperare il contenuto di una variabile nell'array superglobale `$_REQUEST` (che contiene a sua volta anche `$_GET`, `$_POST` e `$_COOKIE`). Questo metodo controlla ed elimina l'eventuale codice HTML o PHP contenuto nella variabile e restituisce il contenuto ripulito.

Noterete anche che la variabile viene stampata sullo schermo utilizzando:

```
<?php echo $variabile?>
```

 invece che il vecchio `<?=$variabile?>`.

Questo permetterà al nostro script di essere utilizzato su tutti i server e di non provocare problemi di compatibilità.

2.2 Che cosa è un Plugin

Una delle caratteristiche più entusiasmanti di Joomla è rappresentata dalle sue funzionalità che possono essere estese all'infinito, attraverso l'installazione di componenti aggiuntivi e Plugin.

I componenti sono grossi programmi in grado di gestire funzionalità molto complesse e solitamente possiedono le loro famiglie di Plugin e moduli che svolgono determinate parti del lavoro. Quasi tutte le funzionalità speciali dei vari siti sono gestite da componenti accuratamente testati nel corso degli anni.

I componenti possono svolgere le funzioni più svariate, come ad esempio la gestione della newsletter, oppure la gestione di prenotazioni per un albergo.

Nelle prime versioni di Joomla i Plugin erano chiamati Mambot.

I Plugin sono piccoli programmi in grado di svolgere funzioni altamente specializzate come ad esempio occuparsi di ottimizzare il sito per la sua indicizzazione nei motori di ricerca, oppure creare gallerie di immagini con effetti speciali.

I Plugin sono degli script, delle funzionalità, che vengono attivati in particolari situazioni, detti eventi, che vengono scatenati dal CMS.

Un Plugin è un componente esterno che estende le “capacità” native di un software e viene attivato da particolari eventi: ad esempio quando si verifica un evento predefinito, e in questo caso, i Plugin che sono associati all’evento stesso vengono eseguiti; pertanto, i Plugin, sono un ottimo modo per estendere le funzionalità di Joomla, senza alterarne il cuore.

Questo significa che, mentre Joomla prepara il contenuto da visualizzare, il nostro plugin aggiunge qualcosa ai preparativi.

Joomla possiede diversi tipi di plugin, che allo stesso tempo, sono anche i nomi delle sotto directory dove possiamo trovare i plugin stessi, o delle sotto cartelle dei siti web dove i file plugin sono memorizzati.

Come già detto in precedenza c’è molto nel pacchetto base di Joomla.

Ulteriori componenti, moduli, plugin e lingue sono elencate nella cartella estensioni di Joomla (Joomla extension directory JED).

Oltre a queste parti ben note, Joomla è composto anche di parti meno conosciute come librerie e pacchetti che non trovano un posto centrale all’interno del progetto Joomla. La ragione di questo è molto semplice: le librerie, sono sviluppate al di fuori del progetto di Joomla e usate da altri progetti open source; mentre i template sono qualcosa di molto individuale e sono, pertanto, disponibili sui siti dei loro progettisti.

Nell’area di gestione estensioni nel pannello amministrativo di Joomla si può abilitare, disabilitare e disinstallare le estensioni, facendo funzionare tutto proprio come un interruttore, semplicemente accendendo o spegnendo l’interruttore dell’estensione.

L’installazione di un Plugin avviene tramite Gestione Estensioni, nel backend del nostro sito Joomla.

2.2.1 Diverse tipologie di Plugin

Andiamo ora a spiegare nel dettaglio quali sono e a cosa si riferiscono i vari tipi di Plugin presenti in Joomla:

- authentication: in questa tipologia sono inseriti i plugin utilizzati per estendere il processo di autenticazione degli utenti nel framework Joomla.

Questi tipi di plugin consentono agli utenti di autenticarsi da varie fonti, ed offrono quindi una grande flessibilità e potenza al sistema.

L'autorizzazione è il processo per specificare i diritti di accesso. È preceduto da autenticazione, che si verifica quando il primo tentativo di essere autorizzato fornisce le corrette credenziali. Ci si deve autenticare con il proprio nome, utente e password, così da essere autorizzati a diventare un membro di un gruppo di autorizzazione.

Joomla offre tre possibilità per effettuare l'autenticazione. Si deve essere in possesso di almeno un plugin di autenticazione abilitato, altrimenti si perderanno tutti gli accessi al sito;

- editors e editors-xtd: sono i plugin che implementano o estendono le funzionalità dell'editor di Joomla.

Joomla di base, è dotato di due editor: per poter così impostare l'editor predefinito per il proprio sito web. Si può inoltre assegnare un editor diverso per ogni account utente.

I quattro plugin presenti in editor-xtd generano i pulsanti sotto la finestra dell'editor;

- search: sono plugin usati per estendere le funzionalità di ricerca che consentono agli utenti di trovare contenuti diversi nel database del sito web Joomla.

La ricerca di questi plugin può essere effettuata tramite specifiche categorie;

- system: sono utilizzati per l'esecuzione di un codice in vari momenti del caricamento del frame work.

I plugin di sistema sono profondamente integrati nella struttura di Joomla e, di solito, si riferiscono al comportamento dell'intero sito;

- user: utilizzati per estendere le funzionalità degli utenti e legati a specifiche funzioni;
- extension: l'estensione di tipo plugin è collegato alle attività relative alla gestione delle estensioni di Joomla;
- finder: questi sono i plugin per la ricerca "classica" del componente di Joomla. Questo plugin è utile per i contenuti, link web, contatti, categorie, e newsfeed e tutti questi, insieme, implementano la sua funzione di ricerca.

Questi plugin devono essere attivati se si cerca di ottenere risultati di ricerca dalle rispettive sezioni. Se si vuole cercare dei componenti aggiuntivi, si deve

avere le disponibilità dei rispettivi plugin;

- captcha: è un programma che è in grado di identificare se l'utente è un essere umano o un computer.

I captcha vengono usati da molti siti web per prevenire gli abusi da "bots" o programmi automatici, di solito scritti per generare spam; infatti nessun programma è in grado di leggere il testo distorto così come gli esseri umani; questo fa sì che i bots non possono navigare nei siti protetti dai captcha;

- quickicon: i plugin di questo tipo controllano gli aggiornamenti per Joomla e per le estensioni di terze parti installate e avvisa quando si sta visitando il sito tramite pannello di controllo;
- content: i plugin di questa tipologia forse sono i più visibili agli utenti.

Si occupano, infatti, di effettuare controlli o inserire funzionalità all'interno dei contenuti: per inserire, all'interno di un contenuto, una gallery o un tab o un campo, spesso si usano plugin di tipo content.

Oltre al contenuto dei plugin di Joomla, tutti i contenuti dei plugin sono legati al testo che viene inserito in un articolo.

Quest'ultimo tipo di plugin è quello scelto e utilizzato per la realizzazione del sistema di votazione del sito web USD San Zaccaria, a cui questa tesi fa riferimento. I Plugin Content sono plugin che vengono eseguiti quando si verificano eventi di contenuti.

Il funzionamento di un plugin è trasparente all'utente che naviga sul sito mentre per quanto riguarda gli amministratori, sono trasparenti solo alcuni tipi di plugin mentre altri devono essere avviati singolarmente.

I plugin content vengono attivati all'interno del file `view.html.php` situato in `Joomla/ components/com_content/views/article` nella classe `ContentViewArticle`.

Quando si crea un nuovo plugin è inoltre importante che il nome del plugin non sia in conflitto con nessuno dei altri plugin già presenti.

Quando viene scritto un plugin di contenuti in Joomla, ci sono diversi eventi a cui è possibile eseguire il codice.

Ogni evento plugin può essere attivato in diversi punti del setup articolo e per questo esistono vari eventi.

2.2.2 Eventi di un Plugin Content

Joomla fornisce per ogni tipo di plugin, eventi predefiniti, per esempio gli eventi contenuto:

- `onContentPrepare`: questa è la prima fase nella preparazione dei contenuti per l'uscita ed è il punto più comune per i contenuti plugin orientati a fare il loro lavoro.

L'evento `onContentPrepare` viene attivato prima che l'articolo sia stampato sullo schermo. Mentre questo plugin modifica l'articolo, gli altri eventi aggiungono contenuti a questo articolo;

- `onContentAfterTitle`: si tratta di una richiesta di informazioni che dovrebbe essere posta tra il titolo e il contenuto del corpo dell'articolo. Permette di aggiungere contenuti anche dopo la stampa del titolo ma prima che venga visualizzato l'introtex;
- `onContentBeforeDisplay`: si tratta di una richiesta di informazioni che dovrebbe essere posta immediatamente prima del contenuto generato;
- `onContentAfterDisplay`: si tratta di una richiesta di informazioni che dovrebbe essere inserita subito dopo il contenuto generato;
- `onContentBeforeDelete`: è un evento che si richiama prima che il contenuto dell'articolo venga cancellato;
- `onContentAfterDelete`: è un evento che si richiama subito dopo la cancellazione del contenuto dell'articolo;
- `onContentBeforeSave`: è un evento che si richiama prima che il contenuto venga salvato nel database;
- `onContentAfterSave`: è un evento che viene chiamato dopo aver salvato il contenuto nel database.

Tra tutti questi eventi sopra descritti `onContentBeforeDisplay` è stato scelto e utilizzato per lo sviluppo del sistema di votazione.

CAPITOLO 3

CREAZIONE DI PLUGIN CONTENT PER JOOMLA

3.1 Creazione dei file necessari per il Plugin Content

Per poter creare un Plugin è necessario costruire la cartella principale, nella quale poi, andremo a salvare tutto il materiale necessario per lo sviluppo del Plugin.

Questa cartella, a lavoro finito, servirà per installare il Plugin e Joomla in automatico la salverà nelle sue suddivisioni delle cartelle.

Dal momento che utilizziamo un Plugin Content, Joomla salverà questa cartella in “plugin/content”.

La creazione di un Plugin si sviluppa in diverse fasi che andremo ora ad illustrare nello specifico.

3.1.1 Creazione del file XML

Tutti i Plugin Content di Joomla contengono un file XML.

XML (Extensible Markup Language) è un linguaggio di markup, ovvero un linguaggio che può essere utilizzato per descrivere altri linguaggi, definendone gli elementi e la grammatica.

Un linguaggio di markup è un linguaggio che consente di descrivere dati attraverso dei marcatori o tag, e si propone di integrare, arricchire e, nel lungo periodo, sostituire html come linguaggio di markup standard per il World Wide Web.

XML è un linguaggio di marcatura che deriva dal linguaggio SGML (Standard Generalized Markup Language). Entrambi sono linguaggi di marcatura testuale, ossia, come già detto, inseriscono all'interno del documento delle etichette o marcature (tag), che permettono di descrivere il contenuto del documento stesso.

A differenza di HTML, XML non ha però tag predefiniti e non serve per programmare né per definire pagine Web, ma esclusivamente per definire altri linguaggi.

In realtà, XML di per sé non è altro che un insieme standard di regole sintattiche per modellare la struttura di documenti e dati. Questo insieme di regole, dette più propriamente specifiche, definiscono le modalità secondo cui è possibile crearsi un proprio linguaggio di markup. Le specifiche ufficiali sono state definite dal W3C (World Wide Web Consortium).

La struttura di questo file è simile a quella dei file XML degli altri tipi di estensioni. Il file XML è denominato con lo stesso nome del file PHP (altro file richiesto per la creazione del Plugin).

Per scrivere il file XML si inizia sempre con il tag “xml” che definisce con l’attributo “version” la propria versione, e con l’attributo “encoding” il formato in cui è scritto, di solito UTF-8.

Esempio:

```
<?xml version = "1.0" encoding = "utf-8"?>
```

Con il tag “extensions” si definisce il tipo di estensione e all’interno di questo tag troviamo altri attributi:

- “version” che specifica la versione di Joomla per cui è stato realizzato il Plugin;
- “group” che specifica di che tipo di Plugin si tratta;
- “content” che specifica in quale cartella andrà inserito.

È anche possibile creare un Plugin che non risponda agli eventi base di Joomla, ma, in questo caso l’attributo non sarà quello delle categorie esistenti.

L’attributo “method” indica invece a Joomla la modalità di installazione dell’estensione. Ad esempio scegliendo method = “upgrade” al tag extensions, il Plugin può essere installato senza disinstallare una versione precedente; tutti i file esistenti verranno sostituiti, e i vecchi file non verranno eliminati.

La fase successiva è l’aggiunta di informazioni sul Plugin e su chi e quando l’ha scritto, attraverso gli appositi tag elencati qui sotto:

- “name”: definisce il nome del plugin;
- “creationDate”: definisce la data di creazione del plugin;
- “author”: definisce il nome della persona che ha creato il plugin;
- “authorEmail”: definisce l’indirizzo email della persona che ha creato il plugin;
- “copyright”: definisce il detentore del Copyright;
- “version”: definisce la versione del plugin;
- “license”: definisce la licenza con cui viene distribuito il plugin;
- “description”: definisce la descrizione del plugin e delle sue funzionalità.

Una volta inseriti i dati descrittivi del Plugin, tramite l'apposito tag "files", includiamo altri file che serviranno per lo sviluppo del Plugin; utilizziamo all'interno di questo tag ulteriori tag:

- "filename": indicano all'installer di Joomla i file del modulo da copiare nella relativa cartella. Come per le estensioni di Joomla solo i file e le cartelle presenti nell'elenco saranno installati nel Plugin; inoltre se nel file XML sono inseriti file non presenti della directory di installazione, il sistema non installerà il Plugin e restituirà un messaggio di errore;
- "folder": definisce la cartella in cui sono inseriti i file; questo perché alcuni file come ad esempio file .js o file .css, hanno bisogno della creazione della propria cartella all'interno della directory dei Plugin, altrimenti non vengono riconosciuti e questo causerebbe un errore durante l'installazione del Plugin.

Esempio:

```
<files>
  <filename plugin="votazione">votazione.php
</filename>
  <folder>images</folder>
  <folder>file</folder>
</files>
```

Per quanto riguarda l'internazionalizzazione, si usano i file di lingua che vengono installati in administrator/languages/, (in Joomla sono già installati codici di linguaggio). Per inserirlo nel file XML si utilizza l'apposito tag "language".

Esempio:

```
<languages folder="language">
  <language tag="it-GB">it
GB/plg_content_nameofplugin.ini</language>
</languages>
```

Si potranno aggiungere, all'interno del file XML, parametri di configurazione attraverso gli appositi tag:

- "config": contiene la configurazione del nostro Plugin;

- “fields” : indica l’insieme dei parametri definiti per il plugin, che sono configurabili dall’utente dal backend e utilizzabili all’interno del plugin stesso;
- “fieldset” : serve per raggruppare più campi;
- “field” : rappresenta il singolo parametro disponibile per il plugin.

All’interno del tag “field” si possono utilizzare diversi attributi:

- “field name” o (nome campo): per indicare il nome del parametro;
- “label” o (etichetta): è il nome del parametro visualizzato nella configurazione del Plugin, nella Gestione Plugin;
- “description” o (descrizione): è il testo di aiuto che appare quando l’utente appoggia il cursore del mouse sopra l’etichetta del campo;
- “default” o (predefinito): è l’impostazione predefinita per un determinato parametro, cioè se l’utente non immette o seleziona un valore per il parametro, è possibile impostare un valore predefinito;
- “type” o (tipo di campo): indica a Joomla in che modo disegnare l’input. Se si ha familiarità con i moduli HTML, ci sono vari modi per ottenere input: si può scegliere tra diversi tipi di parametri, le caselle di controllo, pulsanti di opzione, o aree di immissione del testo.

Quando si aggiungono i parametri di configurazione di un Plugin di Joomla, ci sono vari tipi di input da utilizzare:

- a. tipo radio: è un pulsante di opzione;
- b. tipo lista di parametri: è un parametro che fa riferimento a un elenco a discesa per permettere all’utente di poter scegliere;
- c. tipo input: è un tipo di input di testo, che permette di creare una casella di testo in cui poter scrivere.

Questi parametri possono essere suddivisi in tipi di parametri diversi, come le opzioni di base e le opzioni avanzate, tramite il tag “fieldset” e assegnando rispettivamente il nome basic e advanced.

Per il nostro Plugin abbiamo aggiunto un parametro di configurazione (chiamato noblog) che permette di visualizzare la parte di votazione dei palloni nel singolo articolo o anche nella sezione news.

Esempio:

```
<config>
  <fields name="params">
    <fieldset name="basic">
      <field name="noblog" type="radio" class="btn-
group" default="0" label="Nascondi la visualizzazione della
votazione nelle news" description="Non permettere la
visualizzazione dei voti anche nella sezione news.">
      <option value="1">SI</option>
      <option value="0">NO</option>
    </field>
  </fieldset>
</fields>
</config>
```

Si termina il file XML con l'apposito tag "extension" che era stato aperto all'inizio del file.

Dopo aver creato il file XML lo si salva con il nome votazione.xml e si copia all'interno della cartella principale del Plugin creata precedentemente.

Per poter verificare il tutto, si entra con le proprie credenziali nella parte backend di Joomla, nella sezione Gestione Estensioni – Plugin. Aprendo il nostro Plugin troveremo al suo interno i nostri parametri scritti nel file XML (vedi figura 3.1).

Tutti questi parametri, per poter essere utilizzati, saranno richiamati dal file PHP.

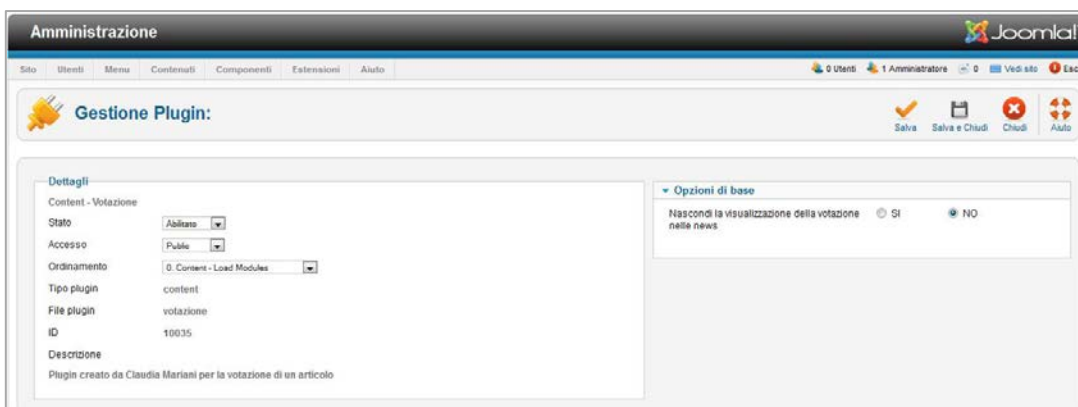


Figura 3.1 – Plugin visualizzato dal backend di Joomla

3.1.2 Creazione del file PHP

Il file PHP è il file principale, quello che permette di eseguire tutto il lavoro.

All'interno di questo file verrà inserito il codice che si occuperà di svolgere la funzione per cui il Plugin è stato creato.

Per prima cosa si inizia a scrivere il file PHP con la licenza generale e con le informazioni relative all'autore del plugin di contenuti e, si introduce una riga di codice, ad esempio: `defined('_JEXEC') or die ('Accesso riservato');` che impedisce alle persone di accedere direttamente al file PHP.

Successivamente inseriamo la dichiarazione della classe che implementerà le funzionalità del nostro plugin.

La classe `plgContentVotazione` estende la classe `JPlugin` da cui eredita tutti i suoi metodi.

Per indicare che si sta scrivendo un plugin Content, si aggiunge il seguente codice:

```
class plg<PluginType><PluginName> extends JPlugin {
    public function __construct(&$subject, $params ) {
        parent::__construct( $subject, $params );
    }
}
```

- `PluginName`: è il nome del plugin (Votazione);
- `PluginType`: è il tipo di plugin (Content);
- `plgTypeNameofPlugin`: è il nome della classe (`plgContentVotazione`).

Classe Plugin ha al suo interno il costruttore che richiama il padre costruttore della classe (classe `JPlugin`) e include la funzione di gestore di eventi, cioè viene richiamato quando si verifica un determinato evento.

Va inoltre fatta molta attenzione all'uso delle maiuscole che fanno parte del codice ricordandosi di non lasciare spazi.

È necessario sostituire `Nameofplugin` con il nome del proprio plugin che sarà `plgContentPlugin`.

A questo punto si può partire con la programmazione del Plugin e lo sviluppo delle sue funzioni:

Il primo evento scritto è `onContentBeforeDisplay` che serve per visualizzare il contenuto votazione prima del contenuto dell'articolo, cioè implementa l'omonimo evento che viene scatenato in fase di preparazione del contenuto.

Per prima cosa bisogna verificare che il contesto della pagina sia quello legato al blog o ad un singolo articolo, (tramite l'apposita variabile `com_content`) e, nel caso si verifichi questo, si memorizza l'id dell'articolo stesso e si controlla il parametro `noblog`. Se è diverso da uno, setta la view di pagina come articolo ed esegue il tutto su di un singolo articolo e alla fine richiama la funzione `ContentVotazione`.

Questa funzione parte azzerando il valore della variabile `conteggio`, cioè il numero di persone che hanno dato un voto, poi tramite una query recupera i dati di votazione relativi al nostro articolo.

Se questa produce risultati allora si tiene traccia dei valori ottenuti dalla somma dei voti e dal conteggio del numero di voti. Successivamente si verifica la posizione dell'utente cioè si controlla se la persona ha eseguito il login di autenticazione.

A questo punto si chiama la funzione `plgContentVotazioneBall` che produce html relativo alla grafica passando i vari parametri.

La funzione `plgContentVotazioneBall` inizia caricando il file CSS (Cascading Style Sheets) responsabile dello stile e in base all'autenticazione dell'utente vengono prodotti due output grafici diversi: se l'utente è loggato, viene caricato il file javascript necessario alla routine di votazione e a questo punto si può calcolare il voto medio. Il calcolo inizia creando tutto il markup html di stampa inserendo i valori di voto medio e il numero di voti dell'articolo e termina restituendo l'output stampato.

Al termine della creazione del file, lo si salva con il nome `votazione.php`, sempre all'interno della nostra cartella del Plugin.

3.1.3 Creazione del file `index.html`

HTML è l'acronimo di Hypertext Markup Language o Linguaggio di contrassegno per gli Iper testi.

HTML è il linguaggio di markup solitamente usato per la formattazione di documenti ipertestuali disponibili nel World Wide Web sotto forma di pagine web.

È un linguaggio di pubblico dominio la cui sintassi è stabilita dal World Wide Web Consortium (W3C), e che è basato su un altro linguaggio avente scopi più generici.

L'HTML non è un linguaggio di programmazione perché non prevede definizione di variabili, strutture dati, funzioni e strutture di controllo, ma solamente un linguaggio di formattazione che descrive, cioè, le modalità di impaginazione o

visualizzazione grafica del contenuto, testuale e non, di una pagina web. È più propriamente definito un linguaggio di contrassegno o di marcatura, che permette di indicare, attraverso degli appositi marcatori detti “tag”, come disporre gli elementi all’interno di una pagina e supporta l’inserimento di script e oggetti esterni quali immagini o filmati.

La formattazione consiste nell’inserimento, all’interno di un testo, di marcatori o etichette, chiamati appunto tag, che descrivono caratteristiche come la funzione, il colore, le dimensioni e la posizione che avranno all’interno della pagina.

L’HTML è, quindi, come già detto, il linguaggio attraverso cui si può indicare la disposizione dei vari elementi che vanno disposti in una pagina Web. Un documento html non è nient’altro che un file di testo con, ad esempio, indicazioni sul colore delle scritte, sulla posizione delle immagini all’interno della pagina e su come far scorrere il testo.

Il linguaggio HTML, pur essendo dotato di una sua sintassi, non presuppone la logica ferrea e inappuntabile dei linguaggi di programmazione: se ci si dimentica di chiudere un tag, ad esempio, non verranno prodotti dei messaggi di errore; se non rispettate la sintassi probabilmente non otterrete la visualizzazione della pagina che desiderate, ma nient’altro.

Il file index.html, viene inserito per impedire che un utente possa vedere la lista dei file contenuti nella directory stessa. Il contenuto può essere il seguente:

```
<html><body bgcolor = "#FFFFFF"></body></html>
```

In caso contrario non è necessario inserire alcun codice ma è sufficiente crearlo.

3.2 Aggiunta di ulteriori file necessari per lo sviluppo del Plugin Content

Fin ora abbiamo descritto i tre file essenziali per la creazione del Plugin, andiamo adesso ad illustrare i file aggiuntivi e facoltativi per lo sviluppo del Plugin.

3.2.1 File CSS

L’acronimo CSS sta per Cascading Style Sheets (fogli di stile a cascata) e designa un linguaggio di stile per i documenti web.

I CSS istruiscono un browser o un altro programma utente su come il documento debba essere presentato all’utente, ad esempio definendone i font, i colori, le immagini di sfondo, il layout, il posizionamento delle colonne o di altri elementi

sulla pagina, ecc.

I CSS servono per gestire tutto il layout di un sito Web. Grazie ad essi è possibile intervenire sulla formattazione del testo o di documenti HTML, XHTML e XML, sul posizionamento degli elementi grafici e sulla disposizione che questi elementi avranno rispetto a diversi media e device.

L'introduzione del CSS si è resa necessaria per separare i contenuti dalla formattazione e permettere una programmazione più chiara e facile da utilizzare, sia per gli autori delle pagine HTML che per gli utenti, garantendo contemporaneamente anche il riuso di codice ed una sua più facile utilizzo.

Le regole per comporre il CSS sono contenute in un insieme di direttive emanate a partire dal 1996 dal W3C.

Il primo passaggio è creare il foglio di stile CSS.

Per il nostro Plugin è stato creato il file CSS con il nome miostile.css e salvato nella cartella File all'interno della nostra cartella del Plugin.

Dopo aver creato il file CSS, è necessario aggiornare il file XML del Plugin per includere il file CSS. Tramite il tag "folder" si include la cartella chiamata file, in cui è inserito il file CSS appena creato denominato miostile.css.

In ultimo si carica il foglio di stile utilizzando la funzione addStyleSheet, all'interno della funzione plgContentVotazioneBall con questa riga di codice:

```
$document->addStyleSheet  
(JURI::base()."plugins/content/votazione/  
file/miostile.css");
```

Come si può notare, in primo luogo abbiamo dichiarato \$document e quindi abbiamo usato il \$document->addStyleSheet per aggiungere l'URL del foglio di stile al documento. Poiché questo file CSS è ora incluso all'interno del Plugin, possiamo ottenere l'URL assoluto del Plugin utilizzando una combinazione di JURI::base() insieme con il percorso del file che inizia con plugins/content.

Se effettivamente l'aggiunta del file è andata a buon fine, verrà aggiunto nel file index di Joomla la riga di codice:

```
<link rel="stylesheet"  
href="http://YourDomain.com/plugins/content/votazione/  
miostile.css"  
type="text/css"/>
```

3.2.2 File Javascript js

Javascript è un linguaggio di scripting orientata agli oggetti e comunemente usato nella creazione di siti web e programmazione web, interpretato e sviluppato da Netscape.

Javascript, ad esempio, viene utilizzato nelle pagine web per aggiornare automaticamente e in tempo reale la data, aprire la pagina linkata in una finestra pop-up e per cambiare l'immagine o il testo a seconda delle azioni svolte attraverso il mouse.

Il Javascript è il linguaggio di scripting più utilizzato.

Per il nostro Plugin è stato creato il file javascript con il nome votazione.js e salvato nella cartella file all'interno della nostra cartella del Plugin.

Sono poi stati aggiornati i file XML del Plugin per includere il javascript votazione.js tramite il tag "folder" perché il file è salvato all'interno della cartella file.

In ultimo carichiamo il file javascript utilizzando addscript.

Nella funzione plgContentVotazioneBall viene usata la seguente funzione per aggiungere il file:

```
$document->addscript  
(JURI::base())."plugins/content/votazione/file/  
votazione.js");
```

si può usare JURI::base() per contribuire a generare l'URL completo per il nostro file javascript, essendo esso stesso presente all'interno della cartella del nostro plugin.

Appunto per questo Joomla aggiunge la seguente riga di codice nel file index di Joomla:

```
<script  
src=http://YourDomain.com/plugins/content/votazione/  
file/votazione.js
```

Questo può servire anche a verificare, durante la progettazione, se il file javascript è stato inserito nel nostro plugin.

All'interno contiene una funzione che non fa altro che prendere i dati sul voto che l'utente vuole esprimere e spedirli via ajax al server.

Se il server produce un esito positivo viene aggiornata la parte di votazione con un messaggio di ringraziamento, in caso contrario avvisa che l'articolo è già stato votato.

3.2.3 File *ajax.php*

In informatica AJAX, acronimo di Asynchronous JavaScript and XML, è una tecnica di sviluppo software per la realizzazione di applicazioni web interattive. Lo sviluppo di applicazioni HTML con AJAX, si basa su uno scambio di dati in background fra web browser e server, che consente l'aggiornamento dinamico di una pagina web senza esplicito ricaricamento da parte dell'utente.

AJAX è asincrono, nel senso che i dati extra sono richiesti al server e caricati in background senza interferire con il comportamento della pagina esistente. Normalmente le funzioni richiamate sono scritte con il linguaggio JavaScript. Tuttavia, e a dispetto del nome, l'uso di JavaScript e di XML non è obbligatorio, come non è necessario che le richieste di caricamento debbano essere necessariamente asincrone.

AJAX è stato enunciato per la prima volta da Jesse Garrett, nel 18 Febbraio 2005, come titolo di un post all'interno del suo blog.

Non si tratta di una nuova tecnologia, né di un'invenzione, bensì, di un concetto utilizzato per sviluppare applicativi avanzati e particolari quali Gmail, Google Maps o Google Suggest. Il concetto è in parte espresso nell'acronimo scelto, un utilizzo asincrono di Javascript che attraverso l'interfacciamento con XML, può permettere ad un client di richiamare informazioni lato server in modo veloce e trasparente, allargando gli orizzonti delle Rich Internet Applications.

AJAX non è un nuovo linguaggio di programmazione, ma piuttosto l'utilizzo di un insieme di tecnologie già esistenti.

Con AJAX viene poi introdotta l'arte di scambio di dati con un server e l'aggiornamento di parti di una pagina web senza ricaricare l'intera pagina.

All'interno di AJAX c'è una funzione responsabile della gestione di un voto e responsabile del salvataggio al database.

AJAX importa i file di Joomla e così si ottiene il voto dalla richiesta http Post.

A questo punto si verifica che il voto sia compreso tra i parametri uno e cinque, se questo avviene si controlla poi che l'utente non abbia già votato. Se anche questo

controllo ha esito positivo, si aggiunge l'utente votante nella lista dei voti effettuati e memorizzati nella tabella `content_controllo`. Infine aggiorna il voto dell'articolo corrispondente nella tabella `content_votazione`.

3.3 Gestione del database in Joomla

Joomla memorizza i dati plugin e parametri nei propri database utilizzando un client MySQL, come phpMyAdmin.

I dati Plugin vengono memorizzati nella tabella `_extensions` e Joomla utilizza un prefisso prima del nome per nominare le varie tabelle.

Il metodo utilizzato per la gestione del database è quello di creare due file SQL, uno di installazione e uno di disinstallazione. Questo perché, nel momento in cui il Plugin viene disinstallato, non rimane nessuna traccia dei dati precedenti e con una nuova installazione del Plugin sarà poi creata una nuova tabella.

Nel file SQL di installazione, le due tabelle che servono per la memorizzazione dei dati vengono scritte usando il linguaggio SQL.

Mentre nel file di disinstallazione, è stata inserita la riga di codice che permette di eliminare la tabella in caso di disinstallazione del Plugin; utilizzando sempre il linguaggio SQL.

L'operazione successiva è quella di aggiornare il file XML andando ad aggiungere le righe di codice che permettono di far funzionare i due file SQL, tramite i tag "install" e "uninstall".

3.3.1 Creazione tabelle

Nel nostro caso sono state create due tabelle: una, chiamata `votazione`, in cui viene memorizzato il numero dell'articolo per poterlo identificare, per segnare il totale dei voti ricevuti e il numero dei votanti; e una seconda tabella in cui viene invece memorizzato un id, il numero dell'articolo e il numero dell'utente che viene usato per controllare che la votazione sia unica da parte di quell'utente e per verificare che sia registrato.

Per prima cosa si deve creare tabella all'interno del database per memorizzare i dati del nostro Plugin. Per fare questo abbiamo utilizzato il generico comando `CREATE TABLE IF NOT EXISTS nome_tabella (definizioni)`, con l'opzione `IF NOT EXISTS`; questo evita che venga visualizzato un errore: ad esempio nel caso la

tabella fosse già presente, in modo quindi da crearla solo se non esistente.

Nome_tabella indica, ovviamente, il nome che si vuole assegnare alla tabella, e le definizioni, indicano i tipi di colonne di cui si deve comporre la tabella e le caratteristiche.

Si usa il comando PRIMARY KEY (id) per indicare quale colonna sarà la chiave primaria, cioè quella che identificherà univocamente le righe, automaticamente questo aggiunge al campo id la proprietà di unicità; se una riga ha un determinato numero di id, questo implica che nessun'altra riga potrà avere lo stesso id.

La tabella votazione è composta dai campi:

- id_articolo INT(10) NOT NULL: è il campo in cui viene memorizzato il numero dell'articolo, ma non contiene alcuna informazione riguardante l'articolo stesso.

Questo valore, mai visibile all'utente, è un intero composto al massimo da dieci cifre. Un limite ragionevole, però, consente di non sprecare inutilmente spazio per la memorizzazione di un numero eccessivamente grande. Attraverso NOT NULL imponiamo che sia sempre definito e che quindi non possa mai restare vuoto;

- somma INT(15) NOT NULL: è il campo che memorizza il totale dei voti ricevuti;
- conteggio INT(10) NOT NULL: è il campo che memorizza il numero di persone che hanno dato un voto.

La tabella controllo si compone di:

- id INT (10) NOT NULL AUTO_INCREMENT: serve per identificare, attraverso un numero, la tabella.

Dovrà essere un numero intero composto al massimo da dieci cifre e si deve specificare, a MySQL, che questo deve essere un campo AUTO_INCREMENT e che quindi sarà lui che lo incrementerà ogni volta che si inserisce un nuovo record;

- user_id VARCHAR (50) NOT NULL: è il campo destinato a contenere l'id degli utenti, al quale si assegna una stringa di lunghezza massima di 50 caratteri.

Nel caso in cui venga specificata una stringa di tipo VARCHAR si deve sempre impostare anche la lunghezza massima della stringa e anche

l' `user_id` (not null);

- `id_articolo` INT (10) NOT NULL: è lo stesso campo della tabella votazione.

Tabella Votazione:

Campi	Tipo
<code>id_articolo</code>	int (10)
<code>somma</code>	int (15)
<code>conteggio</code>	int (10)

Tabella Controllo:

Campi	Tipo
<code>id</code>	int (10)
<code>user_id</code>	varchar (50)
<code>id_articolo</code>	int (10)

Tabella 3.1: Tabelle utilizzate per il sistema di votazione.

3.3.2 Inserimento record

Questa fase prevede l'inserimento dati tramite il linguaggio SQL e viene implementata nel file `ajax.php`.

Una volta creato un database lo si deve popolare inserendo i record tramite l'utilizzo del comando:

```
INSERT INTO nome_tabella (elenco_campi) VALUES ('elenco_valori');
```

`INSERT INTO` indica che vogliamo introdurre un nuovo elemento nella tabella e la parentesi che segue indica in quali colonne vogliamo specificare il valore da inserire; le colonne verranno poi aggiornate automaticamente da MySQL.

Tramite `VALUES` si elencano i valori che andranno inseriti nelle colonne specificate in precedenza, i quali sono contenuti nella seconda parentesi e disposti nello stesso ordine con cui abbiamo specificato le colonne.

Tutti i valori devono essere indicati fra due apici che possono essere omessi solo nel caso di valori numerici.

Questa tabella non richiede alcuna informazione al database, se non l'avvenuto inserimento.

Nella tabella votazione viene inserito il voto dell'utente, solo se esprime il voto per la prima volta, inserendo anche l'id dell'articolo e aggiornando il campo del conteggio segnalandolo con il numero uno.

Nella tabella controllo inserisce invece l'id dell'articolo e l'id dell'utente per controllando ed evitando che l'utente voti due volte.

3.3.3 Ricavare i dati inseriti nella tabella

Questo genere di query è quella che probabilmente verrà utilizzata più spesso, perché serve per ricavare i dati precedentemente inseriti in una tabella, tramite il comando:

```
SELECT colonne FROM nome_tabella WHERE condizioni opzioni.
```

Con il termine colonne si indica tutte le colonne di cui vogliamo il valore; le condizioni sono quelle che devono verificarsi, perché una riga sia visualizzata e le opzioni ci serviranno, ad esempio, per ordinare le righe.

`SELECT * FROM` è il comando che serve per selezionare una o più righe, in particolare indica che l'interesse è rivolto a tutte le colonne della tabella controllo. Se utilizzassimo solo questa parte della query, tutte le righe verrebbero restituite senza un particolare ordine; questo perché non è stato applicato nessun vincolo alla ricerca.

Per la nostra tabella è stato inserito un vincolo diverso, specificando, tramite `WHERE user_id="$user_id"`, che l'interessamento era rivolto solo agli utenti che hanno un id esattamente uguale al valore contenuto nella variabile `$user_id`. E `id_articolo = "$cid"`: facendo ciò si seleziona solo l'utente e il numero articolo utilizzati nella pagina `index.php`.

3.3.4 Modifica valori

Questa è la funzione attraverso cui si possono modificare alcuni valori di determinate righe:

```
UPDATE nome_tabella SET nome_campo='valore' WHERE  
campo_di_condizione='valore_di_condizione'.
```

Questa notazione è piuttosto generale: `nome_tabella` indica come sempre il nome della tabella, `nome_campo='valore'` indica la modifica da apportare ad una determinata colonna e `campo_di_condizione='valore_di_condizione'` indica le condizioni che si devono verificare in una riga affinché questa possa essere modificata.

Nella nostra tabella votazione, viene poi aggiornato il valore dell'articolo votato in precedenza da altri utenti: si osserva che il campo somma, aumenta in base al valore dei voti totali e si ha l'incremento del campo conteggio dei voti, in base al numero di utenti votanti.

3.4 Funzione di accesso al database in Joomla

L'interrogazione e la manipolazione dei dati contenuti nel database di sistema avviene attraverso i metodi della classe JDatabase. La funzione di accesso si sviluppa in tre fasi:

1. Connessione al database

Quando si sviluppano componenti, moduli o plugins per Joomla normalmente non è necessario eseguire istruzioni di connessione a database in quanto è sufficiente ottenere un riferimento all'oggetto database di sistema, e utilizzare la connessione già aperta ed inizializzata da Joomla. A questo scopo si utilizza il metodo statico getDBO(), sta per DataBase Object, della classe JFactory che è una classe statica, che viene utilizzato per recuperare i riferimenti a molti degli oggetti di sistema.

Tramite l'oggetto database di sistema si possono compiere tutte le normali operazioni di accesso ai dati.

Esempio:

```
$database = &JFactory::getDBO();
```

2. Impostare una query

Prima di poter essere eseguita ogni query deve essere impostata tramite il metodo setQuery().

Esempio:

```
$database->setQuery('SELECT * FROM #__votazione');
```

Il metodo setQuery() sostituisce i caratteri #__ con il prefisso delle tabelle specificato in fase di installazione di Joomla.

È importante utilizzare sempre questa notazione standard quando si impostano query da eseguire sulle tabelle di sistema, in quanto se l'utilizzatore modifica il prefisso predefinito per le tabelle del database di sistema, la query fallirebbe.

3. Eseguire una query di selezione

Per eseguire una query di selezione esistono tante funzioni ma per la nostra, è stata utilizzata la funzione loadObject(), per recuperare una riga dal database cioè per poter ottenere i dati che occorrono per la votazione.

Esempio:

```
$votazione=$db->loadObject();
```

CAPITOLO 4

GESTIONE DEGLI UTENTI

4.1 Tecniche di sicurezza per la votazione

Per poter avere una corretta e reale votazione e per essere sicuri che l'utente possa votare una sola volta, all'interno di un articolo, è necessario effettuare una corretta scelta della tecnica più appropriata da adottare.

Possiamo scegliere fra tre diversi metodi: la tecnica dei cookies, il controllo dell'indirizzo IP e la registrazione dell'utente.

Queste tre metodologie consentono diversi gradi di sicurezza e anche di impegno per l'utente che andrà a votare.

Nella maggioranza dei casi viene impiegato uno dei due primi metodi o una combinazione di questi, questo va a vantaggio della velocità di compilazione per l'utente votante, con l'obiettivo di avere il maggior numero di risposte possibili ma a discapito della reale votazione.

4.1.1 La tecnica dei cookies

I cookie sono un metodo rapido per memorizzare sul computer le informazioni che vogliamo persistano anche nelle successive visite al sito.

Sono molto utili per registrare piccoli dati come ad esempio, il nome dell'utente o una serie di preferenze di navigazione, ma non sono adatti per informazioni critiche come password o dati personali in quanto potrebbero creare problemi di sicurezza.

I cookies sono file di testo di piccole dimensioni, che vengono scritti dal server che ospita il sito sul computer e che viene a collegarsi ad una determinata pagina. Il loro utilizzo, è ormai pratica comune, perché, il cookie, tiene traccia del fatto che, se un utente, visitando una pagina e decidendo di votare un articolo non si trova poi in grado di poterlo votare nuovamente, utilizzando lo stesso pc, perché il secondo voto non verrà accettato.

Il tempo di permanenza del cookie deve essere sufficientemente basso in modo da non impedire ad un secondo utente di poter votare a sua volta dallo stesso pc; se il tempo è troppo alto bisogna collegare una sola persona ad ogni singolo pc, è quindi facile intuire che, nel caso di postazioni pubbliche, questa assunzione non può essere fatta.

Va inoltre considerato che il visitatore più esperto può, o utilizzare diversi browser dallo stesso PC, oppure impostare il proprio browser in modo che non accetti i cookies, negando così la possibilità al sistema di riconoscere il voto doppio. Allo stesso modo, avendo un minimo di competenze tecniche sull'utilizzo dei browser, è possibile rimuovere i cookies dal proprio computer e tornare quindi a votare.

L'utilizzo della tecnica dei cookies è una possibilità interessante, perché, anche in presenza di un IP dinamico, se l'utente si disconnette per poi ricollegarsi ad Internet, il cookie rimarrebbe.

Un loro difetto è che sono facilmente eliminabili e non tutti gli utenti li accettano.

4.1.2 La tecnica del controllo dell'IP (Internet Protocol address)

Indirizzo IP è un'etichetta numerica che identifica univocamente un dispositivo (host) collegato ad una rete informatica.

Questo si sviluppa memorizzando e confrontando gli indirizzi IP del computer dal quale è stata effettuata la votazione: sono però molti i casi dove, a fronte di IP identici, ci possono essere persone diverse che esprimono il proprio voto. Questo accade con tutti i computer che possiedono un IP statico: ovvero quelli collegati permanentemente ad una rete locale, come ad esempio uffici, università, enti pubblici, etc. In questo caso, il rischio è di scartare i voti ricevuti da postazioni pubbliche dove si alternano più persone.

Un altro caso in cui si potrebbero avere indirizzi uguali è quello di computer diversi posti in una rete locale, che accedono ad Internet tramite un server Proxy (computer che garantisce la sicurezza di una rete locale): ad esempio, se nella vostra azienda è presente un Proxy, tutti i voti inviati dai computer della vostra azienda avranno il medesimo indirizzo IP. Viceversa, collegandosi a Internet con un modem da casa, il pc otterrà generalmente un IP assegnato dal provider di accesso a Internet, possibilmente differente a ogni collegamento. In questo modo o lo stesso utente appare al sistema di raccolta voti con un IP differente ogni volta che si ricollega, rendendo quindi inefficace questo metodo di controllo.

La tecnica più usata del controllo IP è basata, sicuramente, sull'indirizzo IP del visitatore. Questo consiste nel registrare assieme alla preferenza anche l'IP ogni qualvolta che si vota; al voto successivo, incrociando gli IP già presenti, se si manifesta un secondo accesso il voto viene negato.

Scegliendo la tecnica del controllo dell'IP si ha, come vantaggio, la semplicità di realizzazione ma, come difetto, il non funzionamento in caso di presenza di un IP dinamico o di una rete esterna.

4.1.3 Tecnica della registrazione dell'utente

L'ultima tra le possibilità più utilizzate è rappresentata dalla registrazione dell'utente on-line: il lettore, una volta iscritto e quindi solo dopo essersi loggato, può esprimere il proprio voto. Questo garantisce che ad ogni voto corrisponde uno e un solo utente iscritto.

Per ogni sistema di sicurezza ci sarà sempre qualcuno o qualcosa in grado di violarne il funzionamento, l'importante è cercare di ridurre al minimo la possibilità di violazione aumentando il livello di sicurezza. Ecco perché, per il sistema di votazione utilizzato per il sito www.usdsanzaccaria.it è stato scelto come controllo di sicurezza la tecnica della registrazione dell'utente (vedi figura 4.1).



The image shows a web page for user registration. At the top, there is a breadcrumb trail: "Sei qui: [Home](#) > [Contatti](#) > [Registrazione](#)". Below this, the title "Registrazione utente" is displayed in red. A note indicates "* Campi richiesti". The form consists of six input fields, each with a red label and an asterisk: "Nome: *", "Nome utente: *", "Password: *", "Conferma password: *", "Indirizzo email: *", and "Conferma indirizzo email: *". At the bottom of the form, there are two buttons: a red "Registrati" button and a red "Annulla" button.

Figura 4.1 – Registrazione Utente nel sito

4.2 Gestione degli utenti in Joomla

La gestione degli utenti è particolarmente importante proprio perché dalla loro partecipazione dipendono la ricchezza dei contenuti e la vivacità dei siti web.

Tra gli utenti di ogni CMS, vanno distinti, i visitatori dagli utenti registrati: i primi hanno la possibilità di navigare liberamente all'interno del sito ma senza accedere ad alcun servizio; mentre i secondi, diventando, appunto, utenti registrati possono

accedere ad ogni parte del sito, come ad esempio effettuare la votazione di un articolo, perché vengono identificati tramite l'accesso che hanno effettuato in precedenza.

Joomla, come ogni CMS, fa una distinzione tra visitatori e utenti registrati con diversi permessi: viene infatti suggerito, ai visitatori che di solito provengono dai motori di ricerca o da siti social media di registrarsi.

All'interno di un sito creato con Joomla, o un altro CMS evoluti, si possono dividere gli utenti in tre categorie: visitatori del sito, utenti registrati ed amministratori.

I visitatori, sono utenti occasionali, che cercano semplicemente informazioni, ma che non collaborano attivamente ai lavori del sito stesso. Possono navigare in tutte le aree pubbliche del sito e a volte lasciare qualche commento.

Gli utenti registrati, possono effettuare l'accesso, ed essere quindi identificati, tramite il processo di login. Il loro accesso prevede aree riservate ed in genere, possono effettuare ulteriori operazioni rispetto ai visitatori.

Gli amministratori possiedono inoltre l'accesso al backend del sito, possono quindi, a diversi livelli, occuparsi della configurazione dell'amministrazione del sito stesso.

In Joomla, per gli utenti registrati ed amministratori viene fatta un'altra suddivisione in livelli tramite un sistema di controllo degli accessi (ACL): ogni utente fa parte di uno dei Gruppi Utente con un determinato livello di accesso e dei permessi stabiliti; allo stesso modo, ogni gruppo, ha dei permessi predefiniti e un livello di accesso.

All'interno del pannello Gestione Utenti è possibile visualizzare alcuni dettagli relativi alle persone registrate al sito e verificarne la loro connessione o abilitazione.

È inoltre possibile usare dei filtri, che si trovano in alto a destra, per poter modificare gli utenti visualizzati, in modo da poter scegliere di visualizzare, ad esempio, solo gli utenti Editor. Questa è un'opzione molto utile, nel caso in cui, la presenza degli utenti registrati fosse davvero molto alta.

Una volta effettuata la registrazione nel sito da parte dell'utente, basta selezionarlo per poter entrare nel pannello di Modifica Utente, all'interno del quale sono presenti i dettagli dell'iscrizione che eventualmente possono anche essere modificati (ad esempio può essere cambiato l'indirizzo e-mail oppure l'user di accesso al sito).

Soprattutto all'interno del pannello Modifica Utente possono essere assegnati permessi speciali agli utenti, come ad esempio i permessi di administrator; ovviamente questa possibilità dipende da chi concede il permesso stesso (ad esempio solo il Super Administrator, può infatti creare, un altro Super Administrator).

La pagina di gestione degli utenti (vedi figura 4.2) è raggiungibile tramite il pulsante Gestione Utenti che si trova nel pannello di controllo o nella voce Utenti all'interno del menù ed è divisa in quattro sezioni:

- Utenti: tramite il quale si possono vedere, creare, modificare tutti gli utenti del sito;
- Note utenti: sono delle informazioni aggiuntive da associare agli utenti e possono essere poi raggruppate in Categorie;
- Gruppi utenti: in cui è possibile creare dei nuovi gruppi di utenti oltre a quelli già esistenti.

I gruppi sono in pratica insiemi di utenti accomunati dal possesso delle stesse autorizzazioni.

I gruppi predefiniti, già presenti nelle versioni precedenti di Joomla, sono: Public, Manager, Administrator, Registered, Author, Editor, Publisher, Super Users.

In Joomla si possono creare un numero infinito di gruppi utente, ai quali può essere assegnato un livello di accesso: in questo modo avremo un utente collegato ad un gruppo che si collega ad un livello di accesso.

- Livelli di accesso: permettono di creare nuovi livelli di accesso e, definire, per ognuno, i gruppi di utenti che potranno avere accesso alla visualizzazione del livello.

I livelli di accesso riguardano i permessi di visualizzazione o meno delle informazioni presenti nel frontend del sito Internet.

I livelli di accesso sono un pacchetto di gruppi che possiedono tre livelli di accesso:

- public: accesso in lettura a tutti contenuti;
- registered: accesso in lettura soltanto per gli utenti registrati;
- special: accesso in lettura consentito ai soli autori o utenti di livello superiore.

Tramite la combinazione di gruppi di permessi e livelli di accesso, è possibile risolvere tutti i casi di utilizzo possibile: ad esempio, in un articolo si può limitare l'accessibilità ad un livello di accesso.

Grazie alla combinazione dei permessi di gruppo e dei livelli di accesso, diventa semplice la gestione di ogni tipo di situazione che ci si presenterà.

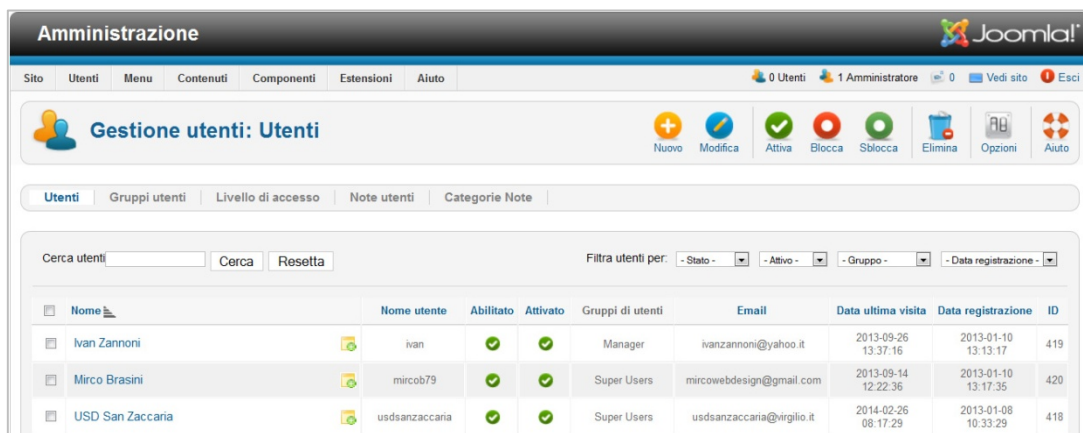


Figura 4.2 – Pannello di Gestione Utenti in Joomla

Il ciclo di vita del primo processo di registrazione nel vostro sito è stato completato con l'installazione di Joomla. La persona che ha installato Joomla ora è il Super Amministratore, si tratta quindi dell'utente con più privilegi all'interno del sito che può effettuare qualsiasi tipo di operazione compresa la creazione di nuovi utenti e nuovi Super Amministratori.

Nelle opzioni degli utenti possiamo settare la configurazione del modulo utente costituito da tre tabelle:

- componente: all'interno di quest'area si può configurare se è possibile o meno effettuare una registrazione utente al sito;
- mail di massa: in questa tabella si possono configurare le impostazioni di una email statica: inviando un email di massa agli utenti;
- permessi: servono per gestire le impostazioni di permessi per ciascun gruppo utente.

Joomla distingue gli utenti registrati in due gruppi: gli utenti del front-end e gli utenti del back-end.

I primi hanno il compito di fornire contenuti al sito, ma è precluso l'accesso alle funzioni amministrative e possono avere quattro livelli di autorizzazione:

- Registered: gli utenti tramite questo primo livello possono proporre la pubblicazione di collegamenti ipertestuali e avere accesso a risorse non disponibili per gli ospiti ma senza creare, modificare o pubblicare contenuti;
- Author: è colui che può scrivere e modificare le proprie informazioni, e inviare nuovi articoli che dovranno essere approvati e pubblicati da un amministratore. È in grado però di modificare solo i suoi articoli;
- Editor: può scrivere e modificare tutte le informazioni e i contenuti che si trovano nel front-end, cioè modificare gli articoli di tutti gli altri utenti, ma non può pubblicarli;
- Publisher: può fare tutto quello che è consentito all'editor e decidere se pubblicare o meno alcuni contenuti e alcuni articoli.

Gli utenti del backend si occupano dell'amministrazione e gestione dell'aspetto grafico e servono per aumentare i privilegi degli utenti.

Gli utenti del back-end sono gli amministratori del sito, cioè coloro che possono accedere al pannello amministrativo. Esistono per questi utenti, tre tipi di permessi:

- Manager: può creare contenuti nel back end, ed ha la possibilità di compiere queste operazioni tramite il pannello di amministrazione, invece che utilizzare il front-end. Il Manager ha accesso a tutte le funzioni di controllo dei contenuti, ma non può modificare la configurazione globale del sito o l'aspetto grafico, né può gestire gli utenti o caricare moduli, componenti e estensioni;
- Administrator (Admin): amministra gli utenti, ma non può modificare la configurazione globale del sito e non può inviare messaggi di massa, né modificarne modelli. Può cambiare i template e il layout del sito, gestire gli utenti di livello uguale o inferiore al proprio;
- Super Administrator: possiede tutti i permessi ed è in grado di fare tutto. È l'unico che può creare un altro Super Administrator e che può compiere le operazioni di amministrazione all'interno del sistema Joomla, compresa quella di gestire, modificare o eliminare altri utenti di ruolo pari al proprio.

4.3 Programmazione del modulo login in Joomla

I visitatori hanno la possibilità di registrarsi al sito tramite un interfaccia, chiamata form, che provvede a raccogliere i dati dell'utente per poi inviarli al database all'interno della tabella utenti e definirne quindi la registrazione.

Joomla offre questa possibilità attraverso un modulo. I Moduli sono delle funzionalità che estendono le normali funzioni di Joomla e che vengono posizionate all'interno del nostro sito; tutti i moduli sono gestiti dal backend nella gestione moduli.

Per poter arrivare alla gestione moduli, si può usare il pulsante "Gestione Moduli" che si trova nella parte centrale del pannello di controllo, oppure il menu in alto, seguendo la voce: Estensioni - Gestione Moduli.

A questo punto il modulo Login (vedi figura 4.2) visualizza un form nome utente e password, mostrando anche un link per recuperare la propria password dimenticata; mentre un altro link verrà visualizzato per abilitare la registrazione automatica per gli utenti e può essere personalizzato e posizionato i qualsiasi pagina del sito.

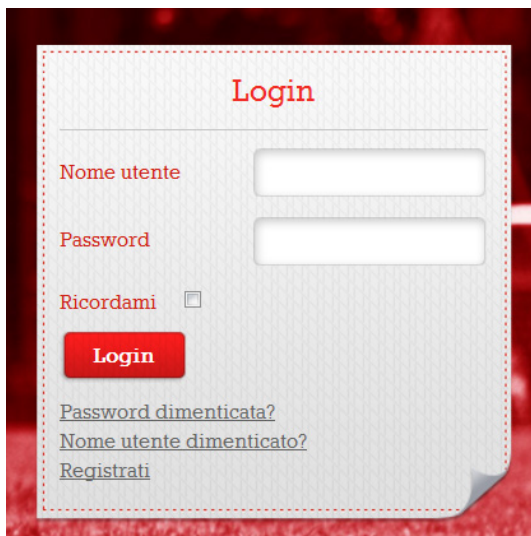


Figura 4.2 – Modulo Login nel sito www.usdsanzaccaria.it .

Il modulo Login viene creato dal Pannello Amministrativo di Joomla; all'interno della sezione Gestione Moduli, tramite il pulsante Crea, si apre una finestra in cui viene selezionato il modulo che interessa e quindi selezionato il login che permetterà di visualizzare la maschera di login per permettere agli utenti di registrarsi.

Nel template del sito USD San Zaccaria, all'interno della cartella dove sono contenuti i file del modulo login, nel file default.php, si deve aggiungere righe di codice per posizionare nella pagina scelta il modulo login.

Una volta creato e posizionato il modulo si passa alla creazione della voce di menu registrazione.

La prima operazione da effettuare è la creazione della cartella registrazione all'interno del template, che avviene attraverso la tecnica dell'override, cioè la sovrascrittura del layout dei moduli e dei componenti standard di Joomla attraverso il template, che permette di rendere più personale la pagina visualizzata. La tecnica del override risulta, molto utile e indispensabile quando Joomla viene aggiornato ad una nuova versione, perché permette di sovrascrivere la pagina di default senza ulteriori problemi e senza ripetere le modifiche sul modulo o componente, perché queste sono già state effettuate attraverso il template.

Si cerca all'interno del template di Joomla, nella cartella components\com_users\views\registration\tmpl il file default.php e viene fatta una copia all'interno del template del sito USD San Zaccaria, cioè all'interno della cartella html, creata precedentemente, dopodiché, si crea un'ulteriore cartella com_users e una cartella registration, al cui interno si posizionerà il file default.php, che è quello che andremo poi a modificare.

Successivamente si crea una apposita "voce di menu": lo faremo da Menu - Gestione Menu - Nuovo Menu, a questo punto gli viene assegnato un modulo menu. Si crea la prima voce di Menu - Login che consente agli utenti già registrati di poter inserire le proprie credenziali.

Per fare in modo che gli stili di layout siano inerenti a quelli usati all'interno di questo sito web, bisogna crearne dei nuovi all'interno del template.css.

Al termine di tutto questo vengono modificati gli stili della pagina aggiungendo parti di codice al file template.css, in modo che, le pagine aggiunte siano conformi allo stile del sito web.

L'ultimo passaggio prevede la creazione di un nuovo livello di accesso (vedi figura 4.3), che permetterà di nascondere il menu con la voce accedi agli utenti già registrati.

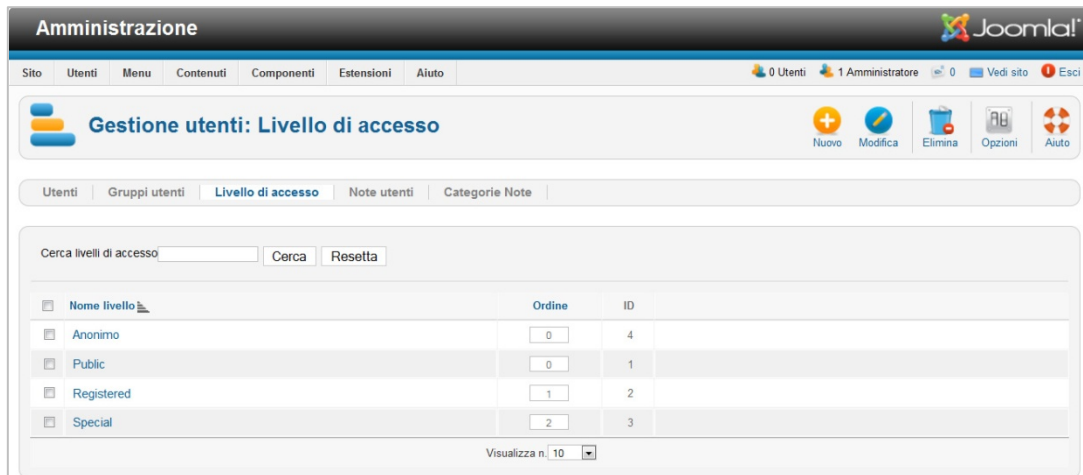


Figura 4.3 – Pannello Livello di Accesso in Joomla

Di default gli utenti registrati potranno vedere tutti i moduli ed i menu che hanno livello d'accesso Public e Registered, in questo modo sarà un pò più complicato gestire la situazione quindi, verrà mostrato un sottoinsieme di moduli o menu solo agli utenti registrati, ed un altro sottoinsieme a quelli non registrati. Sarà quindi creato un gruppo di utenti “Anonimo” che corrisponde a tutti gli utenti non registrati nel sito. La procedura prevede:

1. Creazione di un nuovo Gruppo di Utenti in Gestione Utenti

Questo risulta necessario poiché il gruppo Public esistente è un parente del gruppo Registered: ciò significa che gli utenti del gruppo Registered vedranno Public e Registered; se si crea un nuovo gruppo di utenti sotto Public, gli utenti Registered non vedranno questo gruppo.

Esempio:

- 1) Fare il login nel BackEnd di Joomla;
- 2) Cliccare su Utenti → Gruppi → Nuovo Gruppo;
- 3) Nel Form Gestione utenti: nuovo gruppo utenti inserire i valori seguenti:
- 4) Titolo gruppo = Anonimo;
- 5) Gruppo principale = Public;
- 6) Cliccare su Salva e Chiudi;

2. Creazione di un Nuovo Livello di Accesso in Gestione Utenti

Il nuovo gruppo di utenti ha bisogno di un proprio Livello d'accesso così da poterlo assegnare ai moduli ed ai menu.

Esempio:

- 1) Cliccare su Utenti → Livelli di Accesso → Nuovo livello d'accesso;
- 2) Nel form Gestione Utenti: nuovo livello d'accesso, inserire i valori seguenti:
- 3) Titolo Livello = Anonimo;
- 4) In Gruppi di utenti che hanno accesso alla visualizzazione selezionare solo Anonimo;
- 5) Cliccare su Salva e Chiudi.

3. Modificare il livello di accesso Public in Gestione Utenti

Poiché il vostro sito Joomla è già stato creato con molti moduli e menu con il livello di accesso Public, bisogna permettere agli utenti del gruppo Anonimo di visualizzarli.

Esempio:

- 1) Cliccare su Utenti → Livello di accesso;
- 2) Cliccare ed modificare il livello di accesso Public;
- 3) In Gruppi di utenti che hanno accesso alla visualizzazione selezionare anche Anonimo;
- 4) Cliccare su Salva e Chiudi.

4. Modificare le opzioni di Gestione Utenti

Essendoci, solo per gli utenti non registrati, un gruppo definito, è necessario impostare questo gruppo (creato nel passo 1) come gruppo di appartenenza degli utenti non registrati al posto del gruppo Public.

Esempio:

- 1) Cliccare su Utenti → Gestione Utenti;
- 2) Cliccare sul bottone Opzioni in alto a destra;
- 3) Nella configurazione utenti, selezionare la linguetta Componente ed impostare i valori seguenti:
- 4) Gruppo utenti per visitatori = Anonimo;

5) Cliccare su Salva e Chiudi.

5. Modificare il livello di accesso per i Menu ed i Moduli che dovranno essere visibili solo agli utenti non registrati.

È ora possibile usare la gestione dei moduli per configurare i moduli come richiesta, selezionano i moduli che si desiderano essere visibili solo agli utenti non registrati e cambiare il parametro d'accesso a "Anonimo". La stessa procedura può essere utilizzata tramite la Gestione Moduli perché anche i menù sono configurabili.

CAPITOLO 5

INSTALLAZIONE E FUNZIONAMENTO DEL SISTEMA

5.1 Installazione Plugin

La procedura di installazione è la stessa per tutti i tipi di Plugin e può essere effettuata in tre modalità diverse.

È necessario usare però un solo metodo per installare una determinata estensione.

1. Il primo è caricare un file pacchetto.

Questa scelta viene fatta se è stata creata una cartella poi compressa in un file zip, all'interno del proprio computer locale contenente un'estensione per Joomla. Questo infatti è il metodo utilizzato nel nostro caso.

Dal pannello amministrativo di Joomla in Gestione Estensioni – Installa (vedi figura 5.1), si eseguono i seguenti passaggi:

- Si sfoglia la cartella in cui è stato scaricato il file di archivio dell'estensione;
 - Tramite il tasto “carica file & installa” si carica il file compresso (votazione.zip), Joomla leggerà i contenuti del file di archivio e installerà l'estensione;
 - Se l'installazione è andata a buon fine sarà trasmesso un messaggio “Installazione avvenuta correttamente”, oppure apparirà un messaggio d'errore, nel quale saranno mostrati gli errori effettuati.
2. Se è stato caricato un file zip all'interno di una cartella documenti, nella cartella root, all'interno del server web per Joomla, può essere utilizzata l'opzione installare da cartella per estrarre ed installare la vostra estensione.

Per effettuare questo si seguono alcuni passaggi:

- Creare una cartella temporanea all'interno del disco rigido locale e scompattare il file di archivio dell'estensione nella cartella temporanea;
- Caricare tramite FTP (File Transfer Protocol o protocollo di trasferimento file) i contenuti della cartella, inclusi i file e le sottodirectory, in una cartella sul server;
- Specificare nel campo “installazione dalla cartella” il nome della cartella all'interno della quale sono stati caricati i file e le sottocartelle del pacchetto;

- Cliccare sul pulsante “installa”; e anche in questo caso comparirà il messaggio “installazione eseguita correttamente”, oppure, un messaggio d’errore.
3. Se si conosce l’indirizzo URL di un file zip, che contiene un’estensione per Joomla, questa potrebbe essere un’opzione per scaricare, estrarre ed installare l’estensione.

La procedura prevede che:

- Anziché scaricare il file di archivio nel computer in locale si specifica l’indirizzo URL del file di archivio;
- Si clicca sul pulsante “installa” e Joomla lo installa direttamente dall’indirizzo URL;
- Anche in questo caso sullo schermo compare il messaggio “installazione eseguita correttamente” o messaggio d’errore. Utilizzando questo metodo non si avrà però una copia del file di archivio nel computer.

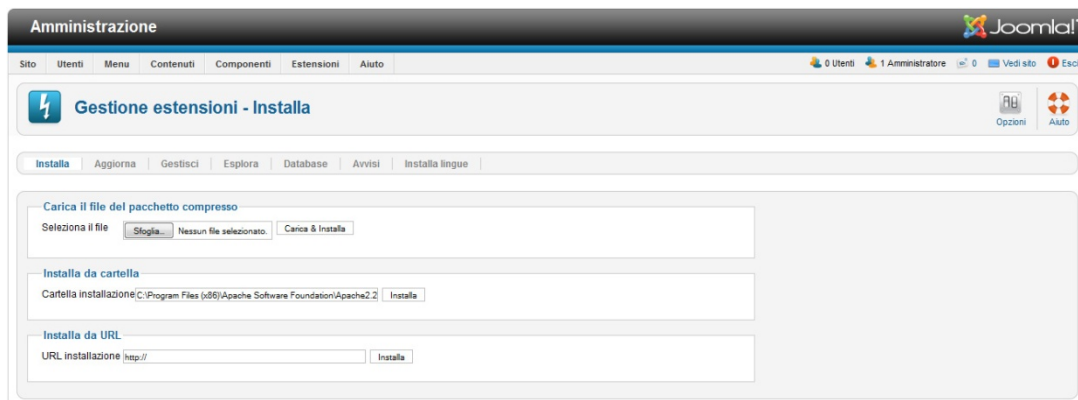


Figura 5.1 – Gestione Estensioni, Installa

5.2 Abilitare e Testare il Plugin

Una volta effettuata l’installazione, si passa all’operazione di configurazione del plugin stesso, che, al momento dell’installazione è disabilitata: sarà quindi importante assicurarsi di abilitare il Plugin.

Se il plugin è stato caricato correttamente, per poter effettuare la configurazione, tramite il pannello amministrativo di Joomla, si seleziona il menu “Estensioni” - “Gestione Plugin”, dove sono presenti tutti i Plugin di Joomla (vedi figura 5.2). A questo punto dovrebbe essere visualizzabile il plugin dal titolo “Content - Votazione”, che viene configurato tramite l’impostazione dei vari parametri di

configurazione e abilitato in modo da renderlo funzionale quando si entra nella pagina web.

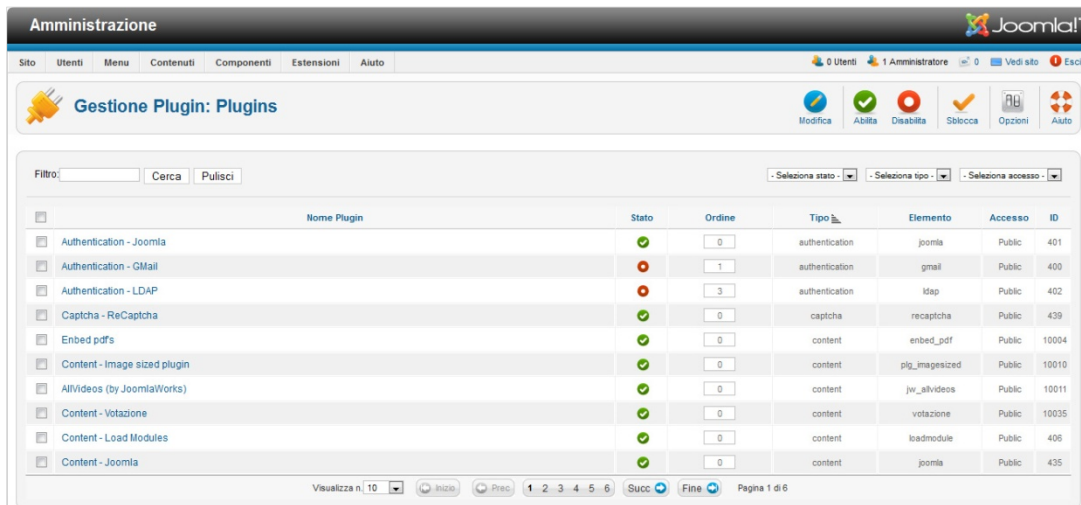


Figura 5.2 – Gestione Plugin, “Content – Votazione”

Per inserire le informazioni necessarie al corretto funzionamento del Plugin ci si deve accertare di aver definito:

- la posizione del Plugin nel template (la denominazione dipende dal template che si sta utilizzando);
- l’impostazione dello “Stato” del Plugin che deve essere “Pubblicato” per un corretto funzionamento del contatore;
- l’impostazione di “Accesso” del Plugin, deve essere “Public”;
- l’impostazione di “Assegnazione plugin” nell’area “Assegnazione menu”;
- il controllo della selezione delle pagine nelle quali si vuole visualizzare la votazione.

Inoltre nelle impostazioni di base del plugin va controllato il parametro noblog, perché permette di mostrare la parte votata sia nel singolo articolo che nella raccolta degli articoli stessi, nella sezione News di serie B femminile (vedi figura 5.3).



Figura 5.3 – Illustrazione votazione nella sezione della raccolta degli articoli.

Per poter terminare la configurazione con successo si deve cliccare sull'icona "Salva e Chiudi"; in questo caso comparirà di nuovo nella lista dei moduli quello con il titolo "Content -Votazione" nello stato attivo (icona verde con spunta).

Connettendosi al sito www.usdsanzaccaria.it, all'interno della parte pubblica, dovrebbe essere visibile la parte di votazione di ogni articolo nella pagina News della sezione serie B femminile (vedi figura 5.4).

In questo modo può essere testato il Plugin e ci si può assicurare del suo corretto funzionamento.



Figura 5.4 – Illustrazione finale del progetto

CONCLUSIONI

In conclusione possiamo pertanto dire che il CMS più diffuso e utilizzato è Joomla. Lo sviluppo di un sito web con Joomla è molto pratico e si ha la possibilità, con le abilità di programmazione, di trasformarlo secondo le proprie esigenze e di renderlo efficiente, sia per la persona che visiterà il sito, sia per colui che sceglie di gestirlo e che non possiede praticità di programmazione informatica.

Nonostante ciò, va ricordato che, progettare un proprio CMS, sarebbe la soluzione migliore da adottare, per poter, di fatto, organizzarlo a proprio piacimento e programmarlo in base alle effettive necessità, senza dover seguire codici o regole predefinite, che si trovano all'interno di un CMS già esistente.

Il sistema di votazione scelto è stato creato tramite un Plugin Content perché il sito scelto si sviluppa in Joomla.

Come metodo di sicurezza utilizzato all'interno della votazione, per fare votare una sola volta l'utente e, per avere una votazione reale, è stato progettato un metodo di registrazione dell'utente. Questo sistema di protezione può essere utile e impiegato anche per altre parti all'interno del sito.

Il vantaggio dello sviluppo di un Plugin, interessa anche altri siti web perché può essere adattata la sua grafica sulla base del contenuto presente all'interno del sito.

Dal momento che il sito www.usdsanzaccaria.it viene organizzato e gestito da me ho avuto la possibilità di controllare il riscontro delle visite relativo agli articoli presenti nel sito e le rispettive votazioni. Si nota quindi che il sistema di sicurezza è funzionale e veritiero per quanto riguarda il conteggio dei voti rispetto ai votanti.

Concludo con un pensiero relativo al mio futuro: vista la mia passione per il calcio e per lo sport in generale, spero di avere la possibilità di lavorare nuovamente allo sviluppo di nuovi siti sportivi.

BIBLIOGRAFIA

1. Bruce H., Castro E., , *HTML 5 e CSS 3*, Tecniche Nuove, 2012
2. Castro E., *XML per il World Wide Web*, Pearson Education Italian, 2001
3. Chimenti R., *Costruire siti dinamici con Joomla! 2.5*, Hoepli, 2012
4. Danchilla B., Gogala M., MacIntyre P., *PHP per professionisti*, Apogeo, 2012
5. Flanagan D., *Javascript: The Definitive Guide, 6th Edition*, O'Reilly Media, 2011
6. Gasston P., *CSS3. Guida completa per lo sviluppatore*, Hoepli, 2012
7. Gigliotti G., *HTML 5 e CSS 3*, Apogeo, 2011
8. Graf H., Kramer J., Mitova M., Radke A., *Joomla! 2.5 Guida per principianti*, Cocate, 2012
9. Marriot J., Waring E., *Joomla La guida ufficiale*, Apogeo, 2013
10. Michele E. Davis & Jon A. Phillips, *Learning PHP & MySQL. Second Edition*, O'Reilly Media, 2007
11. Radtke A., *Joomla! Templates*, Addison-Wesley ,2012

SITOGRAFIA

1. Guida HTML, <http://www.html.it/guide/guida-html/> , (ultima consultazione: Gennaio 2014)
2. Joomla! Documentation, http://docs.joomla.org/Main_Page , (ultima consultazione: Gennaio 2014)
3. Creare Plugin Content, http://docs.joomla.org/J2.5:Creating_a_content_plugin , (ultima consultazione Ottobre 2013)
4. PHP, <http://www.php.net> , (ultima consultazione Ottobre 2013)
5. W3C, www.w3c.it , (ultima consultazione Settembre 2013)
6. Creare Plugin Content, <http://www.inmotionhosting.com/support/edu/joomla-25/create-plugin> , (ultima consultazione Ottobre 2013)

RINGRAZIAMENTI

In questo lungo cammino, senza il sostegno delle persone più care, non ce l'avrei mai fatta a concludere gli studi, perciò, per evidenziare il loro sostegno e la loro importanza, voglio dedicare a loro, un piccolo ringraziamento...

Il ringraziamento più importante e più grande va alla mia FAMIGLIA:

Ai miei GENITORI, che in tutti questi lunghi anni mi hanno sempre sostenuto, hanno sempre creduto in me, senza mai dirmi una parola di troppo e mai un rimprovero, ma solo grandi incoraggiamenti e tanta, tanta fiducia in me.. Grazie, grazie di cuore perché ancora una volta avete dimostrato di essere speciali e meravigliosi.

A mio fratello LORY, che con i suoi rimproveri e le sue affermazioni “Ma datti una mossa!!!” ha saputo tirar fuori da me, ancora di più, quella forza per non mollare mai, per dimostrargli che alla fine ce l'avrei fatta.. Grazie di cuore fratellone. Ti voglio bene!

A mia cognata ENRY, che non mi ha mai detto nulla, ma c'è sempre stata e so che ci sarà sempre.. Grazie!

E poi ci sono gli AMICI:

ARY, la mia sorellina, che ha iniziato con me questo lungo cammino a Bologna, dove proprio in quegli anni più duri, lontano da casa, mi ha aiutato, insegnato tanto e confortato, e che da quel lontano giorno non ha più smesso di sostenermi, starmi vicina e incoraggiarmi giorno dopo giorno e soprattutto c'è sempre stata nei momenti bui, è sempre riuscita a trovare le parole giuste per farmi coraggio e a non farmi mandare all'aria, quello che per me era un grande obiettivo. Soprattutto grazie per questo ultimo periodo, direi proprio che mi sono giocata tutti i crediti che avevo e ora non posso più dirle, o meglio chiederle, nulla ma solo continuare a ringraziarla.. Grazie davvero!

Insieme a lei, la sua famiglia, ELIO, DANIELA E DARIO che oltre all'ospitalità, ai pranzi, cene, merende e tanto altro, mi sono sempre stati vicini come se fossero i miei genitori.. sempre quell'abbraccio e quell'amore che riempie il cuore di gioia e che ti ricorda di mollare mai, e, non l'ho fatto... Grazie di cuore!

Le mie AMICHETTE, le amiche di sempre: ANTO, LAURA, ALE, SARA, SINTO, GEME, SONIA, VANS, ELE, SIMO, DONA, ZANDO.. quelle amiche che ti

prendono in giro perché quella laurea non arriva mai, ma che nello stesso tempo ti sostengono giorno dopo giorno, insomma quelle amiche che nella vita vorresti sempre avere, perché nel momento del bisogno ci sono sempre state, perché ti rimproverano quando stai sbagliando, perché ti fanno crescere, e soprattutto, perché ti fanno capire giorno dopo giorno che ci saranno sempre.. Grazie Grazie Grazie!

Ah a ELE, grazie anche per i suoi caffè e il “mandarino” post - pranzo, sono stati una carica ottimale, per seguire con grande attenzione le lezioni pomeridiane.

VALLY & CRI, i miei amici montanari, che mi conoscono da quando sono piccola e appunto per questo hanno sempre creduto in me perché sanno che in ogni cosa che faccio, in fondo ci voglio arrivare.. Grazie!

Grazie anche ad EZIA E PRIMO.. che ormai mi hanno adottato!!

VIETTA, l'amica incontrata e ritrovata, che in qualsiasi momento e soprattutto nei momenti come questi, quando hai paura che ti capiti un imprevisto, ti dà la tranquillità di esserci sempre, perché basta chiamarla e lei arriva (forse perché ha lavorato al 118!).

GIADA, l'amica lontana, che con le tante chiacchierate e le nostre ore e ore di telefonate, è come se fosse stata a due passi da me..

L'ho sempre sentita vicina.. Grazie!

E poi arrivi tu: ANI, l'amica terremotata, quasi alla fine di questo lungo percorso ma, forse non a caso, nel momento più importante, quando sei quasi ad un passo e hai bisogno, di quello stimolo in più per arrivare prima alla fine; sei riuscita a darmi quell'aiuto in più così importante che mi ha portato dove sono ora. E soprattutto grazie per tutti i vari giri che hai fatto e per la pazienza che hai avuto in questo periodo pre – tesi, tanto stressante.. Grazie davvero di tutto cuore!

Agli amici di Università: Mirco, Vero, Rudy, Luca, Ivan, Giulia, Ale e scusate se dimentico qualcuno, quelli che con me hanno vissuto la nostra università, i laboratori, la sala lettura, la mensa, il famoso caffè della macchinetta, la tensione pre - esame, che ci sono stati anche per lo svago serale, dopo giorni intensi di studio.. Grazie infinite a tutti voi!

A Mirko, un grazie particolare, che mi ha seguito e saputo consigliare al meglio per la stesura della tesi.

Un ringraziamento doveroso va al mio Relatore Dott. Mirko Ravaioli per l'aiuto, la disponibilità e i suggerimenti per la realizzazione di questa tesi.

Di persone da ringraziare ne avrei tante, perché in questi anni ne ho incontrate tante e, proprio perché penso, che ogni persona non arrivi a caso nella vita di ognuno di noi, meritano un ringraziamento anche tutte quelle persone che ho incontrato nella mia carriera universitaria.. GRAZIE A TUTTI!!