

Introduzione alla programmazione per il web

Anno accademico 2013-2014

Relazione conclusiva secondo progetto

Luca Carli – 152011

Davide Lissoni – 151855

Daniele Dellagiacoma – 152027

Introduzione

La creazione del secondo progetto ha ricalcato a grandi linee l'iter percorso nella realizzazione del primo compito assegnato.

Dopo aver capito tutti i task da completare abbiamo deciso di prendere come base per iniziare il primo progetto.

Scelte database

La prima parte del tempo dedicato al progetto è stata trascorsa nella ricerca di una soluzione per apportare delle modifiche al database, rese necessarie da nuovi problemi da risolvere.

Ad esempio alla tabella utente sono stati inseriti i campi email, necessario per la modifica della password, un campo booleano moderatore e un campo per il recupero della password, mentre per i gruppi sono stati aggiunti due campi true/false chiamati pubblico, per indicare se tale gruppo fosse pubblico oppure privato, e chiuso, per indicare se il gruppo fosse stato chiuso da un moderatore.

Scrittura codice

La parte coding è stata suddivisa tra i vari componenti del gruppo ed è stata la parte che ha occupato il tempo della creazione del progetto nella misura maggiore.

Diversamente dal primo progetto, la parte grafica questa volta è contenuta nelle JSP, mentre la parte relativa al codice è presente all'interno delle servlet come nel caso del primo compito assegnato.

La scelta più importante è stata quella relativa alla differenza tra gruppo privato e pubblico, abbiamo deciso che nel momento della creazione di un gruppo non ci siano differenze; mentre nel momento in cui invece si passa da pubblico a privato, l'unico utente rimane sempre l'amministratore, mentre nel caso contrario, ovvero da privato a pubblico, abbiamo deciso di eliminare tutti gli utenti che erano precedentemente all'interno del gruppo.

Servlet:

Le classi del progetto sono suddivise all'interno dei seguenti package:

- Filter – contiene le servlet per gestire i filtri
- DB – contiene le servlet per comunicare con il database
- Listeners – contiene la servlet per aprire la connessione al database
- Servlet – contiene le servlet usate dal processo che non rientrano in nessuna delle precedenti categorie

All'interno di questi package sono contenute le seguenti servlet; nel seguito ne viene descritto il funzionamento, ne viene spiegata la scelta di programmazione effettuata; tra parentesi viene mostrato in quale package sono contenute.

Login(servlet): è la servlet che viene richiamata dal tasto "Login" e controlla che il login effettuato sia corretto, se l'utente sia un moderatore o meno e setta i valori al cookie; finito questo controllo richiama la Home.

MyGroups(servlet): è la servlet che visualizza i gruppi a cui un utente loggato è iscritto (eventuali privati + pubblici) oppure solamente i gruppi pubblici nel caso di utenti che leggono il forum senza aver effettuato il

login.

RecuperoPassword(servlet): è la servlet inviata nel momento in cui

Registration(servlet): è la servlet inviata nel momento in cui viene confermata la registrazione, da qui controlla l'uguaglianza della password e poi, dopo aver inserito nel database il nuovo utente, lo rimanda alla Login.jsp.

HomeLogged(servlet): è utilizzata per caricare tutte le informazioni necessarie alla visualizzazione delle Home.jsp.

CreaGruppo(servlet): dopo aver controllato che il parametro inserito nella TextBox non sia nullo, in tal caso viene mostrata nuovamente la CreaGruppo, viene invocata una query che inserisce il gruppo e la relazione tra lo stesso e il creatore all'interno del database, viene poi visualizzata la ViewGroup.jsp.

InsertPost(servlet): nel caso di parsing effettua le modifiche del caso(QR, indirizzi web, file già caricati) al testo mentre se si caricano uno o più files alla fine del messaggio aggiunge del testo dove mostra tutti i file caricati per quel post e inserisce il post(testo, data e ora, etc) all'interno del database richiama la ViewGroup.jsp.se sei un utente non loggato non potrai inserire alcun post

UploadFile(servlet): crea una nuova cartella "File" nella directory build del progetto, qualora non sia già presente, salva all'interno il file caricato ed inserisce nel database il percorso.

DownloadFile(servlet): cliccando sul nome del file, quest'ultimo viene salvato nella directory predefinita dei download.

AdminSetting(servlet): carica le informazioni visualizzate nella AdminSetting.jsp e la carica

RemoveUser(servlet): modifica il campo "Eliminato" all'interno del record nel database, così da escludere l'utente dalla discussione.

InviteUser(servlet): esegue la query che aggiunge alla tabella "Invito" un record, in attesa della risposta dell'utente invitato ed invia un mail di invito dalla quale si potrà accettare l'iscrizione

Rename(servlet): aggiorna il record relativo al gruppo nel database

ChangeGroup(servlet): controlla se il gruppo è pubblico o privato e lo cambia nel suo opposto

InviteAccepted(servlet): setta il campo booleano "Accettato" come true ed inserisce la relazione tra utente e gruppo .

InviteRefused(servlet): cancella dalla tabella "Invito" il record ed impedisce un nuovo invito.

UserSettings(servlet) rimanda alla jsp UserSetting , ed ha come parametro una stringa di errore nel caso, nel cambio password l'utente abbia inserito qualche dato errato

ModSetting(servlet): puoi vedere la servlet solo se si è un utente moderatore, carica i parametri necessari per creare la tabella che verrà visualizzata nella jsp

CloseGroup(servlet): questa servlet viene visualizzata quando un moderatore decide di chiudere un gruppo, questa esegue una query che sta nel gruppo che dovrà essere chiuso l'elemento booleano chiuso=true, in questo modo nella viewgroup non sarà più possibile inserire nessun post.

NoLoggedEntry: serve per vedere i gruppi pubblici senza effettuare il login, l'utente non loggato ha come id (passato per parametro) il valore 0, da qui viene mandato alla MyGroups.jsp dove vedrà solo i gruppi pubblici.

ChangePassword(servlet): confronta se le due password inserite sono uguali, in tal caso le modifica nel database

ChangePasswordMail(servlet): invia una mail all'indirizzo dell'utente, che con un link può accedere, entro 90 secondi, ad una pagina che gli permette di inserire la sua nuova password

UploadAvatar(servlet): come l'Upload, solamente che crea una cartella "Avatar", nel caso non esista, e controlla che il tipo del file caricato sia JPG.

LogOut(servlet): torna alla ViewLog e chiude la sessione.

DBManager(DB): servlet che contiene tutte le query necessarie

Gruppo-Post-Utente-GruppoMod(DB): servlet che rimappano le tabelle del database.

FilterGroup(filter): chiama una query in cui controlla che l'utente faccia parte del gruppo passato come parametro dall'URL, in caso contrario chiama una pagina presa come attributo della sessione (la pagina precedente).

FilterLogin(filter): controlla se la sessione è attiva e l'utente è loggato, in caso contrario rimanda alla ViewLog.

NoChangeId(filter): chiama una query in cui controlla se l'utente passato come parametro dall'URL sia uguale a un attributo della sessione settato al login, in caso contrario chiama una pagina presa come attributo della sessione (la pagina precedente).

WebappContextListener(listeners): è un listener che ha il compito di accedere e di chiudere il database.

Parte grafica

La parte della grafica, intesa come visualizzazione dei dati e delle informazioni, per questo secondo progetto è stata sviluppata tramite JSP nel linguaggio JSTL.

Come nel primo progetto la parte di grafica relativa all'abbellimento con colori ed impaginazione adeguata non è stata curata dettagliatamente sempre per lo stesso motivo, ovvero sia la mancanza di tempo, sia la relativa utilità nel farlo.

Conclusioni

Le maggiori difficoltà sono state incontrate nella realizzazione del QR, infatti nella pagina proposta c'era un link ad una libreria non più disponibile.

Anche per questa seconda parte di progetto il gruppo si è dimostrato unito ed insieme è riuscito a superare le difficoltà con grande cooperazione.