

Documentazione

Il database contiene:

- i dati anagrafici dell'utente con l'aggiunta di un saldo ore, relativo alle prestazioni offerte e ricevute, e di un campo di sospensione che specifica se l'utente è 'online', può effettuare sia richieste sia offerte, 'offline', non può effettuare nè richieste nè offerte, o 'negativo', non può ricevere prestazioni ma può offrirne;
- i dati relativi alla residenza dell'utente e ai suoi recapiti;
- la messaggistica interna tra due utenti;
- un elenco di categorie e sottocategorie gestite solamente dall'amministratore;
- le prenotazioni effettuate dagli utenti con un campo azione che determina l'accettazione, il rifiuto o la modifica della prenotazione;
- le prenotazioni accettate che diventano prestazioni con l'indicazione di un campo per il punteggio, da 0 a 10, e un eventuale feedback;
- i periodi, ovvero informazioni relative alla data della prenotazione, ora di inizio e di fine della disponibilità dell'utente che ha effettuato la prenotazione;
- un elenco di luoghi dove gli utenti possono ricevere o effettuare le prenotazioni;
- le attività richieste e offerte sono memorizzate in una tabella 'azioni' con l'indicazione dello stato, 'offerta' o 'richiesta'. Nel caso in cui da queste siano state generate delle prenotazioni il campo 'attivo' viene impostato col valore true;
- le informazioni temporali e spaziali relative alle azioni.

Viste realizzate

Sono state realizzate delle viste per facilitare l'accesso ai dati da parte degli utenti.

- info_prestazioni: permette di visualizzare le prestazioni realizzate dagli utenti;
- stat_prestazioni: permette di visualizzare punteggio e feedback delle prestazioni;
- info_prenotazioni: permette di visualizzare le informazioni relative alle prenotazioni, unitamente a chi offre o riceve l'azione, e lo stato della prenotazione (in modifica oppure 'accettata' se è stata generata una prestazione);
- info_illi_prenotazioni: permette di visualizzare le informazioni spaziali e temporali relative alle prenotazioni;
- info_daa_prenotazioni: permette di visualizzare le informazioni relative agli utenti coinvolti nella prenotazione;
- info_action: permette di visualizzare le informazioni relative a tutte le azioni offerte e richieste presenti nella banca del tempo;
- info_accept_action: permette di visualizzare le informazioni relative a tutte le azioni attive, cioè che hanno generato una prenotazione;
- info_indirizzi, info_utenti e anagrafica_clienti: queste due viste ci sarebbero servite per implementare una regola che, lavorando sulle due viste, andasse a inserire i dati dei nuovi utenti nelle tabelle;
- durata_prestaz: permette di visualizzare la durata complessiva di tutte le prestazioni. Essa viene utilizzata all'interno della query per calcolare la percentuale di prenotazioni che si traducono in prestazioni.

Query realizzate

- numero, saldo ore complessivo e valutazione media delle prestazioni: implementata con una vista 'query1'.
NB: questa query restituisce dati significativi solamente se esistono delle prestazioni. Di conseguenza prima di essere lanciata, le prenotazioni in stato di modifica devono essere poste ad 'accettata' in modo che si creino prestazioni da cui estrarre i dati, la media dei punteggi, salvo nuovi inserimenti, risulterà 0;
- percentuale di prenotazioni che si traducono in prestazioni: implementata con una vista 'query2'.
- il volume di prestazioni effettive effettuate in corrispondenza di una certa offerta/richiesta: implementata con una vista 'query3'.

Check realizzati

I check sono stati utilizzati per gestire i campi con valori predefiniti dagli amministratori come ad esempio il sesso, i giorni, il livello, lo stato delle azioni, le azioni delle prenotazioni e il punteggio delle prestazioni . Inoltre sono stati utilizzati come controllo sulla precedenza di date e orari.

Trigger realizzati

- cnt_saldo: si attiva dopo un aggiornamento sul campo saldoH dell'utente e controlla il suo valore, se è minore di -5 il campo sospensione viene posto 'negativo';
- pren_del: si attiva a seguito di una modifica sul campo azione delle prenotazioni e controlla il suo valore, se ha assunto valore 'rifiutata' allora procede con l'eliminazione della relativa tupla sull'entità prenotazioni e pone il valore dell'attributo 'attivo' della relativa azione a false;
- crea_prestazione: si attiva a seguito di una modifica del campo azione dell'entità prestazioni e ne controlla il valore, se tale attributo ha assunto il valore 'accettata' e non esiste nessuna prestazione già attiva per questa prenotazione procede con l'inserimento della nuova tupla nella relazione prestazioni;
- inc_dec_saldo: si attiva dopo la creazione di una prestazione. Esso va ad incrementare e decrementare rispettivamente i saldi delle ore all'utente che ha offerto e a quello che ha ricevuto la prestazione;
- cnt_offline: si attiva dopo una modifica sul campo 'sospensione' della relazione utenti e ne controlla il valore, nel caso in cui il valore sia 'offline' notifica lo stato offline al cliente e per ogni prenotazione in fase di modifica relativa a tale cliente cambia lo stato della prenotazione da 'modifica' a 'rifiutata' in questo modo si attiva il trigger pren_del che si occupa di cancellare la prenotazione, questo perchè per gli utenti offline non è possibile avere delle azioni attive.
- cnt_nega: si attiva dopo una modifica sul campo sospensione dell'utente e ne controlla il valore, nel caso avesse valore 'negativo' ed esistano prenotazioni per azioni richieste da questo utente allora solleva un'eccezione notificando all'utente che ha saldoH negativo e per ogni prenotazione pone il campo azione a 'rifiutata', questo fa sì che si attivi il trigger pren_del che si occupa di cancellare la prenotazione.

Privattizzazione dei dati:

le autorizzazioni dovrebbero essere gestite tramite il comando GRANT tramite il comando all o all privileges è possibile concedere tutti i privilegi utili in lettura e/o scrittura sulla relazione specifica.

Dovremmo specificare l'insieme degli utenti a cui si applica il comando o utilizzare public per estendere i permessi a tutti gli utenti. Gli utenti che ricevono privilegi non possono concederli ad altri ad eccezione del caso in cui venga utilizzata la clausola 'with grant option' che consente la deroga dell'amministrazione dei privilegi oltre alla possibilità di esercitarli limitatamente alla relazione esplicitata insieme al comando.

Assegnando ai singoli utenti il privilegio di 'select' su viste e relazioni è possibile mantenere la privattizzazione dei dati fra gli utenti.

Si dovrebbero gestire i seguenti casi:

- l'utente non può inserire, rimuovere o modificare i nomi delle categorie;
- l'utente non può inserire, rimuovere o modificare i nomi delle sottocategorie;
- l'utente può accedere esclusivamente alle prenotazioni che lo riguardano;
- l'unico utente in grado di poter accettare una prenotazione è colui che riceve la richiesta di prenotazione;
- l'utente può modificare livello, specifica e tipo solo riguardanti le proprie azioni offerte e richieste;

Come utilizzare il database:

- il file di testo sql.txt contiene tutti i comandi sql relativi a : creazione schema, set

search_path, creazione tabelle, viste, trigger e popolamento;

- per testare il trigger di aggiornamento saldo punti (inc_dec_saldo) è sufficiente modificare il campo 'azione' all'interno delle prenotazioni in cui compare 'ani_sar@msn.com' e porlo 'accettata', in questo modo il trigger crea_prestazione creerà la tupla corrispondente alla prenotazione. Nella tabella utenti si potranno osservare le variazioni dei saldi per gli utenti che hanno partecipato alla prestazione;
- per testare il trigger per il controllo del saldo inferiore a -5 (cnt_saldo) è sufficiente cambiare il valore del saldo all'interno della tabella utenti per l'utente 'ani_sar@msn.com' oppure creare azione, prenotazione e prestazione per questo utente; dopodiché il campo sospensione di questo utente assumerà valore 'negativo'.
- Per testare il trigger che elimina una prenotazione (pren_del) è sufficiente porre il campo azione della tabella prenotazioni 'rifiutata'.
- quando si vuole creare una prenotazione bisogna controllare che esista un'azione attiva a cui riferirla e l'utente associato all'azione deve comparire tra gli utenti coinvolti nella prenotazione;
- NOTA: tutti gli id devono essere inseriti dall'utente (non sono auto increment);