

Documentazione Tecnica

Schema relazionale base di dati

Tipo(tipo, descrizione)

Ruolo(ruolo)

Torneo(ID, nome, edizione, data, ora, min_giocatori, max_giocatori, numero_gare, tipo, organizzatore)

Utenti(ID, nome, cognome, email, username, password, ruolo, abilitato, credito)

Iscrizione(utente, torneo_id, punteggio, posizione, prezzo, premio, approvato)

Gara(ID, casa, trasferta, giornata, torneo, risultato, vincitore)

Gara_torneo_libero(ID, torneo, giornata, utente, risultato, punti)

Spiegazione del codice

Ecco la lista dettagliata di tutti i file all'interno del progetto:

- ./classes/include/connect.inc.php
- ./classes/include/MYSQL.class.php
- ./classes/MUMatch.class.php
- ./classes/MURegistration.class.php
- ./classes/MURole.class.php
- ./classes/MUTournaments.class.php
- ./classes/MUType.class.php
- ./classes/MUUsers.class.php
- ./create.php
- ./css/index.css
- ./css/style.css
- ./database/matchup.sql
- ./database/matchup_from_zero.sql
- ./enroll.php
- ./home.php
- ./index.php
- ./login.php
- ./logout.php
- ./match.php
- ./Matchup.inc.php
- ./rank.php
- ./register.php
- ./session.php
- ./tournaments.php
- ./update.php
- ./users.php

Di seguito una spiegazione per ogni file del codice scritto.

Nel file connect.inc.php sono definiti i parametri di connessione al database.

Nel file MYSQL.class.php è definita una classe MYSQL che gestisce tutto quello che riguarda l'interrogazione del database. *Funzione __construct* costruisce un oggetto di tipo MYSQL, *funzione*

connect crea una connessione al database, *funzione queryOk* stabilisce se il risultato della query è andato a buon fine oppure no, *funzione lastInsertInQuery* restituisce l'ID dell'ultimo inserimento fatto, *funzione query* esegue la query, *funzione lastInsert* è uguale alla funzione *lastInsertInQuery*, *funzione sqlInjection* ritorna il valore passato privo di ogni possibilità che avvenga una sql injection, *funzione fetchAssoc* restituisce l'array come coppie di chiave-valore, *funzione fetchArray* restituisce l'array, *funzione numRows* restituisce il numero di righe della query eseguita, *funzione resultsInArray* restituisce tutti i valori della query in un array chiave-valore, *funzione close* chiude la connessione col database ed infine la *funzione destruct* che viene chiamata quando si distrugge l'oggetto creato e chiude la connessione.

Nel file *MUMatch.class.php* si trovano funzioni per gestire le partite. *Funzione construct* costruisce un oggetto e si connette al database, *funzione createMatches* crea tutte le partite dato l'id del torneo e il tipo del torneo, *funzione getAllMatchesFromTournament* restituisce tutte le partite che appartengono al torneo specificato come parametro, *funzione getRank* restituisce il punteggio dell'utente all'interno del torneo specificato come parametro, *funzione updateScore* aggiorna il punteggio e il risultato dopo che una partita è stata giocata, *funzione destruct* chiude connessione col database.

Nel file *MURegistration.class.php* si trovano funzione per gestire l'iscrizione ad un torneo. *Funzione construct* crea un oggetto e si connette al database, *funzione getUsersEnrolledInTournament* restituisce tutti gli utenti iscritti al torneo passato come parametro, *funzione registerUserToTournament* iscrive l'utente al torneo entrambi sono passati come parametro, *funzione updatePrice* aggiorna il prezzo che deve pagare l'utente per poter partecipare al torneo, *funzione updateAward* aggiorna il premio che vince l'utente in base alla classificazione finale nel torneo, *funzione updateApproved* aggiorna lo stato dell'iscrizione cioè se un utente può partecipare o no, *funzione destruct* chiude connessione col database.

Nel file *MURole.class.php* si trova una funzione per prendere tutti i ruoli che gli utenti possono assumere all'interno dell'applicazione. *Funzione construct* crea un oggetto e si connette al database, *funzione getAllRoles* restituisce tutti i ruoli tranne quello dell'amministratore, *funzione destruct* chiude connessione col database.

Nel file *MUTournaments.class.php* si trovano funzioni per gestire i tornei. *Funzione construct* costruisce l'oggetto e si connette al database, *funzione createTournament* crea un torneo con i parametri specificati, *funzione getAllTournaments* restituisce tutti i tornei che ci sono all'interno del database, *funzione getTournamentsUserIsEnrolledIn* restituisce tutti i tornei a cui ha partecipato o sta partecipando un utente, *funzione getTournamentsUserIsNotEnrolledIn* restituisce tutti i tornei a cui l'utente non è iscritto, *funzione getAllTournamentsByOrganizator* restituisce tutti i tornei creati dall'organizzatore specificato come parametro, *funzione destruct* chiude connessione col database.

Nel file *MUType.class.php* si trovano le funzioni che gestiscono i tipi di tornei. *Funzione construct* crea un oggetto e si connette al database, *funzione getAllTypes* restituisce tutti i tipi di tornei, *funzione destruct* chiude connessione col database.

Nel file *MUUsers.class.php* si trovano funzione per gestire il singolo utente. *Funzione construct* crea un oggetto e si connette al database, *funzione isUserAlreadyRegistered* controlla se l'utente è già registrato, *funzione isUserAccepted* controlla se l'utente è stato accettato per accedere all'applicazione, *funzione getAllUsers* restituisce tutti gli utenti che sono registrati nell'applicazione, *funzione loginUser* fa accedere un utente all'applicazione, *funzione getInformationAboutUser* restituisce tutte le informazioni sull'utente passato come parametro, *funzione changeEnableState* cambia il fatto se un utente è abilitato ad accedere all'interno dell'applicazione o no (di default tutti gli utenti devono essere accettati), *funzione getCredit* restituisce il credito accumulato dall'utente (il credito al momento dell'iscrizione è pari a 100), *funzione changePassword* che permette di cambiare la password, *funzione chageName* che permette di cambiare il nome, *funzione changeSurname* funzione che permette di cambiare il cognome, *funzione registerUser* registra un

utente all'interno del database pronto per essere accettato dagli amministratori, *funzione destruct* chiude connessione col database.

Nel file index.css sono state inserite tutte le regole css per renderizzare la pagina iniziale, mentre per tutte le altre pagine viene usato il file style.css.

Nel file matchup.sql sono specificate tutte le funzioni e le tabelle che appartengono al database, l'unica differenza fra questo file e il file matchup_from_zero.sql è che il primo contiene anche alcune tabelle già popolate, mentre il secondo è per una installazione da zero. All'interno del file sono presenti le seguenti funzioni: aggiorna_posizione la quale aggiorna la posizione dell'utente dopo che è stata inserita una gara per far questo calcola il nuovo punteggio dal vecchio; crea_torneo_eliminazione_diretta il quale dato l'id del torneo genera il primo turno del calendario per il torneo, prendendo prima tutte le combinazioni dalle squadre iscritte e poi inserendo le varie gare nella tabella gara tramite la funzione inserisci_valori; crea_torneo_italiana il quale dato l'id del torneo genera tutte le gare del torneo compreso l'andata e il ritorno, il funzionamento è molto simile alla funzione crea_torneo_eliminazione_diretta; inserisci_risultato inserisce il risultato nel database e inoltre imposta il vincitore nella gara nel caso sia un pareggio il campo vincitore rimarrà vuoto e poi aggiorna la posizione nel database; inserisci_valori inserisce i valori delle gare all'interno della tabella gara. La tabella gara è stata fatta in questo modo perchè rende molto più chiara la visione di partita in quanto sulla stessa riga ci sono sia la squadra che gioca in casa che quella che gioca in trasferta; in contrapposizione alla tabella sopra indicata vi è la tabella Gara_torneo_libero nella quale, dato che la modalità del torneo viene scelta dall'organizzatore, le varie gare vengono scelte da lui e possono quindi parteciparvi più giocatori. La tabella Iscrizione associa un utente ad un torneo e vengono comunicati dati riguardanti l'iscrizione al torneo; trigger aggiorna_valori il quale aggiorna il credito dell'utente; tabella ruolo contiene solamente una lista di tutti i ruoli; tabella tipo contiene una lista di tutti i tipi di tornei; tabella torneo contiene il nome del torneo, l'edizione, data ora, numero dei giocatori e organizzatore; trigger aggiorna_edizione che in automatico se verifica che un torneo con quel nome è già stato creato aggiorna l'edizione in modo da poter mantenere una storia; trigger ins_org_in_iscr inserisce in automatico l'organizzatore del torneo alla lista degli iscritti al torneo; tabella utenti memorizza alcuni dati sull'utente tra cui lo username e la password e il credito.

Nel file create.php viene creato il torneo dati ovviamente i parametri per la funzione.

Nel file enroll.php permette all'utente di iscriversi ad un torneo cliccando semplicemente sul pulsante "Iscriviti".

Nel file home.php viene visualizzato il profilo con cui uno accede e inoltre c'è la possibilità di modificare alcune informazioni riguardanti il profilo stesso.

Nel file index.php viene visualizzata la pagina iniziale dove ci si può registrare o accedere.

Nel file login.php viene processata la risposta alla richiesta di login da parte dell'utente.

Nel file logout.php viene distrutta la sessione permettendo così l'uscita dall'applicazione.

Nel file match.php vengono create le gare o in modo automatico oppure in modo manuale in base al tipo di torneo.

Nel file Matchup.inc.php vengono incluse tutte le classi php per avere un unico file da includere in tutti gli altri e implementata la *funzione MUDebug* per debuggare ogni singola variabile all'interno del codice.

Nel file rank.php viene visualizzata la classifica aggiornata e in ordine dal primo in classifica all'ultimo con i relativi punteggi.

Nel file register.php viene processata la risposta alla richiesta di registrazione da parte dell'utente.

Nel file session.php viene creata la sessione e questo file viene incluso in tutti gli altri.

Nel file tournaments.php vengono visualizzati i tornei in base al proprio ruolo all'interno dell'applicazione.

Nel file update.php viene processata la risposta alla richiesta di cambio di alcuni campi all'interno della tabella utente.

Nel file users.php vengono visualizzati tutti gli utenti che sono iscritti al torneo che viene passato alla pagina.

Elenco Prodotti Software

Per realizzazione dell'applicazione sono usati i seguenti prodotti software:

- apache2 per la parte del web server
- php come linguaggio di programmazione lato server, il quale gestisce tutta la logica dell'applicazione
- mysql come dbms relazionale
- phpmyadmin come interfaccia grafica per usare il database mysql

Queste applicazioni sono state installate utilizzando il pacchetto fornito da <https://www.apachefriends.org> cioè XAMPP.

Linguaggi Programmazione Usati

Per quanto riguarda i linguaggi di programmazione ecco una lista dettagliata:

- html per la formattazione dei file a livello del client
- css per migliorare la formattazione dei file html
- php per la logica dell'applicazione
- sql per la parte che riguarda il database, quindi prendere i dati dal database ed inserire i dati
- sql per tutta la parte delle store procedure

Schema concettuale ER della base di dati (in allegato)