

Diario di lavoro

Luogo	Trevano
Data	18.10.2019

Lavori svolti

Oggi ho iniziato ricontrollando la pagina di amministrazione e mi sono reso conto che era piuttosto scomoda da utilizzare con i bottoni che facevano apparire le tabelle, ho quindi deciso di rimuoverli e lasciare le tabelle visibili. Queste sono limitate a mostrare al massimo 7 righe per pagina così che chi visualizza la pagina non debba scrollare troppo per trovare il record che cerca.

Dopo aver ristrutturato il frontend ho continuato con quello che mi mancava, ho iniziato creando la funzione che richiama la pagina "gestione" che mostra in un form l'utente / grotto da modificare. La pagina è un insieme delle pagine di aggiunta di un grotto e di registrazione di un utente e andrà a richiamare i metodi che verificheranno le modifiche e le renderanno effettive (questi metodi sono ancora da fare). Fatto ciò sono passato a scrivere le funzioni di approvazione di un grotto e di eliminazione di un elemento sia nel controller che nel model.

A seguire prima la parte della funzione di eliminazione situata nel controller che verifica che tipo di campo si vuole eliminare, se questo esiste e, nel caso degli utenti, che ci sia sempre almeno un admin dopo l'eliminazione.

...

```
if($type == 'grotto'){
    $id = filter_var($im->checkInput($id), FILTER_SANITIZE_NUMBER_INT);
    $grotto = (new DBConnection)->getGrotto($id);
    if($grotto != null){
        (new DBConnection)->delete($type, $id);
    }
}elseif ($type == 'utente'){
    $id = filter_var($im->checkInput($id), FILTER_SANITIZE_EMAIL);
    $utenti = (new DBConnection)->getUsers();
    $utente = (new DBConnection)->getUser($id);
    if($utente != null){
        foreach($utenti as $item){
            if($item['nome_ruolo'] == 'admin' && $item['id'] != $utente['id']){
                (new DBConnection)->delete($type, $id);
            }else{
                array_push($errors, "Deve sempre esserci almeno un admin");
                $_SESSION['errors'] = $errors;
            }
        }
    }
}
```

...

Fatto ciò ho impiegato il tempo restante nella lezione per testare il corretto funzionamento del codice scritto.

Problemi riscontrati e soluzioni adottate

Oggi ho riscontrato un problema durante l'eliminazione di un campo dalla tabella, infatti cercavo di eliminare un utente o un grotto ma in entrambi i casi non mi era

consentito dato che essi erano inseriti nella tabella **voto** come foreign key. Per risolvere il problema ho dovuto aggiungere al database dove dichiaro le chiavi esterne l'operazione da seguire in caso di eliminazione.

```
create table voto(  
  email_utente varchar(50),  
  id_grotto int,  
  voto float not null,  
  primary key(email_utente, id_grotto),  
  foreign key(email_utente) references utente(email) on delete cascade,  
  foreign key(id_grotto) references grotto(id) on delete cascade  
);
```

Come si può vedere sopra la parte in grassetto fa sì che all'eliminazione del campo a cui si fa riferimento vengano eliminati tutti i figli. Quindi in questo caso succederà che:

- Se si elimina un grotto tutti i voti ad esso verranno persi
- Se si elimina un utente tutti i voti effettuati da esse verranno persi

Questo ho dovuto farlo anche nella tabella foto per i grotti.

Punto della situazione rispetto alla pianificazione

Sono in anticipo sulla pianificazione.

Programma di massima per la prossima giornata di lavoro

La prossima giornata voglio terminare la pagina di modifica di un utente / grotto.