

Diario di lavoro

Luogo	Canobbio
Data	12.11.19

Lavori svolti

Nella lezione di oggi mi sono occupato dei compiti assegnati dal docente responsabile durante la scorsa lezione. Dunque, per cominciare mi sono recato a commentare tutto il codice scritto fino ad ora. I file con più commenti sono quelli principali e quelli su cui per adesso ho lavorato di più, ovvero: "index.php", "login.php", "signup.php", "server.php" e "verify.php". Finito di commentare il codice, ho sistemato il problema del filtro della volta scorsa. Il problema consisteva in questo: la tipologia può essere di 3 tipi (Albergo, B&B, Camping). Il primo alloggio, di tipologia "Albergo", possiede l'id "1". Quando andavo a inserire un filtro per la tipologia, nel caso che il filtro fosse "Albergo", mi veniva mostrato solo il primo alloggio. Nei casi invece che il filtro fosse "B&B" o "Camping", non mi veniva mostrato nessun alloggio. Per sistemare il problema, mi sono accorto che andavo a controllare se la query "SELECT id FROM alloggio WHERE id = \$i AND nome_tipologia = '\$value' LIMIT 1" producesse un risultato, e nel caso che non lo producesse, andavo ad interrompere completamente il for tramite un "break". Quello che mi serviva era invece un "continue", che mi permettesse di terminare la seguente iterazione del ciclo e passare a quella successiva.

```
//Ciclo che viene eseguito in base al numero di alloggi presenti e che stampa gli alloggi.
for ($i = 1; $i < $max_id + 1; $i++) {
    //Se viene utilizzato un filtro per la tipologia utilizzo una certa query.
    if (isset($value)) {
        $accomodation_query = "SELECT id FROM alloggio WHERE id = $i AND nome_tipolo
gia = '$value' LIMIT 1";
        $stmt = $db->prepare($accomodation_query);
        $stmt->execute();
        //Se eseguendo la query viene trovata una riga, prepare una nuova query.
        if ($stmt->rowCount() > 0) {
            $id = implode($stmt->fetch(PDO::FETCH_ASSOC));
            $accomodation_query = "SELECT * FROM alloggio WHERE id = $id LIMIT 1";
        }
        //Altrimenti interrompo la corrente iterazione e passo alla prossima.
        else {
            continue;
        }
    }
    //Altrimenti uso la query di "default".
    else {
        $accomodation_query = "SELECT * FROM alloggio WHERE id = $i LIMIT 1";
    }
    $stmt = $db->prepare($accomodation_query);
    //Eseguo la query.
```

Cambiando il contenuto dell'else, da "break" a "continue", il filtro funziona correttamente.

In seguito, sono andato a creare due nuovi file “credenzialiDatabase.php” e “emailBody.php” che sono i due file che devo modificare ogni volta prima di caricare online il sito. In entrambi i file basta scommentare il codice per lavorare online e commentare quello per lavorare localmente.

Ho anche creato un file “.htaccess” nella DocumentRoot del progetto (ovvero la cartella “one-column-listing”), in cui ho inserito il seguente codice:

```
RewriteEngine On
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule . /errore.php [L]
```

Questo codice fa sì che se l’utente inserisce un URL inesistente, ad esempio “localhost/aaa.php”, egli viene reindirizzato alla pagina “localhost/errore.php”

Successivamente, sono andato a caricare su GitHub, come richiesto, il file “.mpp” contenente il gantt.

Al termine della lezione, prima di scrivere il diario sono andato a caricare il sito sul server (cosa che devo fare ogni martedì sera).

Problemi riscontrati e soluzioni adottate

Nessun problema riscontrato

Punto della situazione rispetto alla pianificazione

In ritardo rispetto alla pianificazione

Programma di massima per la prossima giornata di lavoro

Nella prossima giornata di lavoro dovrei lavorare sulla documentazione, in particolare sui test case. Inoltre, potrei procedere con l’implementazione dei filtri.