

Diario di lavoro

Luogo	Canobbio
Data	20.09.2019

Lavori svolti

Come pianificato ieri oggi ho terminato la pagina di login inserendo un nuovo di controllo ed una nuova feature.

Quando un utente esegue un login ora il sistema controlla se l'utente è effettivamente un professore utilizzando LDAP. Se il sistema rileva che è un professore controlla se quel determinato professore è stato abilitato all'utilizzo dell'applicazione.

Se il professore non è abilitato all'utilizzo del tool viene mostrato un messaggio a schermo nella pagina di login:

Sembra che il tuo account non sia abilitato per accedere a questo servizio, contatta l'amministratore

Subito dopo ho iniziato a lavorare su un sistema che permette il controllo dei permessi dell'utente. Come prima cosa ho creato una classe chiamata *PermissionManager*, la quale permette di leggere i permessi dell'utente dal database. I permessi vengono formattati e messi in una classe chiamata *Permissions* la quale ha delle funzioni che semplificano l'identificazione di un determinato permesso:

```
m ↗ canCreareUtenti():bool  
m ↗ canEliminareUtenti():bool  
m ↗ canPromozioneUtenti():bool  
m ↗ canVisionePrenotazioni():bool  
m ↗ canInserirePrenotazioni():bool  
m ↗ canCancellazionePrenotazioniPrivate():bool  
m ↗ canCancellazionePrenotazioniGlobali():bool
```

Quando l'utente effettua il login correttamente il sistema crea una variabile di sessione chiamata "permissions" la quale contiene al suo interno un oggetto di tipo "Permissions". Per esempio, se mi loggo con il mio account ("luca.dibello", il quale è stato impostato con permessi di amministratore) questo è quello che contiene l'oggetto:

```
C:\Users\luca6\Google Drive\Repo\CPTMeetingRoomScheduler\src\application\views\home\index.php:28:  
object(Permissions)[2]  
  private 'PERMISSIONS_NAME' => string 'admin' (length=5)  
  private 'creazione_utenti' => boolean true  
  private 'eliminazione_utenti' => boolean true  
  private 'promozione_utenti' => boolean true  
  private 'visione_prenotazioni' => boolean true  
  private 'inserimento_prenotazioni' => boolean true  
  private 'cancellazione_prenotazioni_personali' => boolean true  
  private 'cancellazione_prenotazioni_altri_utenti' => boolean true
```

Andando a vedere il codice per la connessione al database ho notato un errore molto grosso: le credenziali di accesso (username, password, database ed host) non erano delle costanti globali ma delle stringhe:

```
DB::$user = "root";
```

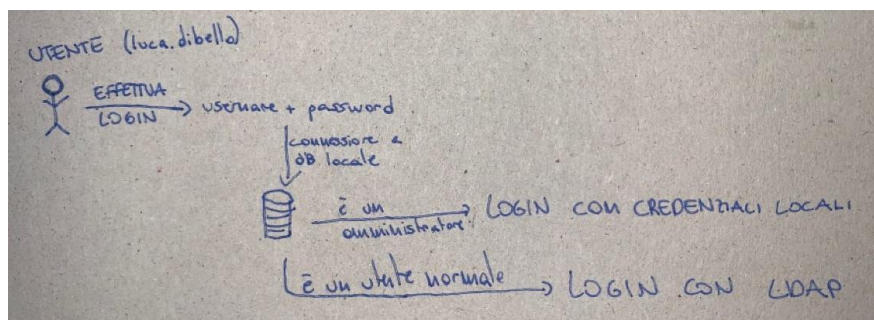
Il problema di questo approccio è che quando bisognerà cambiare server e/o database non sarà molto pulito. Per mettere apposto ho deciso di creare nel file *config.php* delle costanti:

```
define('DATABASE_USERNAME', 'root');  
define('DATABASE_PASSWORD', 'root');  
define('DATABASE_NAME', 'cptmrs');
```

Che vengono poi utilizzate nel file *index.php* per il setup di MySQL:

```
// setup database variables  
DB::$user = DATABASE_USERNAME;  
DB::$password = DATABASE_PASSWORD;  
DB::$dbName = DATABASE_NAME;
```

Questo è uno schemino che ho fatto per illustrare come il login di un utente viene effettuato:



Problemi riscontrati e soluzioni adottate

Oggi non ho riscontrato alcun tipo di problema, tutti i lavori sono stati molto semplici ed oltre a qualche errore di sintassi non ho avuto alcun problema.

Punto della situazione rispetto alla pianificazione

Sono leggermente in anticipo rispetto alla pianificazione.

Programma di massima per la prossima giornata di lavoro

Finire sistema di controllo permessi ed implementare il sistema relativo alle password per i