

GESTIONE CASINÒ | Diario di lavoro - 05.04.2019

Matan Davidi, Thor Döblin, Matteo Forni, Carlo Pezzotti, Mattia Toscanelli

Trevano, 5 aprile 2019

Lavori svolti

Matan ha proseguito con la documentazione del progetto, aggiungendo i capitoli 1.6.3, Progettazione, 1.6.4, Implementazione, 1.6.6, Documentazione di progetto e 6.2, Considerazioni personali.

Thor e Matteo oggi hanno cercato un modo per far andare i test di selenium sul server, creando una virtuale di prova del server e seguendo i passaggi mostrati in questa guida che sono andati a buon fine:

<https://tecadm.in.net/setup-selenium-chromedriver-on-ubuntu/>

Tuttavia per installare chrome si è usata un'altra guida siccome il capitolo riguardante quell'operazione non andava a buon fine:

<https://linuxize.com/post/how-to-install-google-chrome-web-browser-on-ubuntu-18-04/>

Successivamente si ha provato a utilizzare i test di prova tuttavia senza successo, siccome la libreria installata non era nella stessa posizione del file Java.

Carlo ha proseguito con la stesura del codice backend. Durante lo stand-up ci siamo chiariti su un paio di cose che precedentemente erano sconosciute:

- la prima era che pagina mostrare una volta eseguito il login -> la soluzione è stata quella di mostrare la pagina dove vengono mostrati le possibili interazioni che l'utente può fare (modificare la password, le informazioni e visualizzare le promozioni).
- la seconda era la gestione dell'accesso da parte di un utente, ovvero -> se un utente vuole essere ricordato allora le sue info vengono salvate in un cookie (email) con la vita di 7 giorni. Altrimenti vengono usate le session con lo stesso principio.

Il codice che ho sviluppato oggi non è nulla di che. Mi sono occupato di gestire la pagina di profilo di un utente.

```
<?php
require_once "php/loader.php";
session_start();
if(empty($_SESSION["username"])){
    setcookie("error","Pagina non trovata :(", time() + 1000,"/");
    header("Location: error.html");
    exit();
}

$queryRepose = $db->executeQueryWithoutFetch("select * from user where email = '".$_SESSION['username']."'")->fetch();
?>
```

```
<p><span class="header-modify">Nome: </span><span><?php echo $queryRepose["name"]?></span></p>
<p><span class="header-modify">Cognome: </span><span><?php echo $queryRepose["surname"]?></span></p>
<p><span class="header-modify">Nascita: </span><span><?php echo $queryRepose["birthday"]?></span></p>
<a href="modifyInformation.php"><input type="button" class="form-control" value="Modifica Dati"></a>
```

Oggi Mattia ha creato la pagina delle sale, dei giochi e delle promozioni. Tutte e tre hanno lo stesso formato, perché vengono ogni volta che si vuole aggiungere una sala, un gioco o una promozione si deve adattare dinamicamente grazie al pannello che dispone l'admin per le aggiunte di quest'ultima. Ogni volta che l'admin aggiunge per esempio una sala, nella pagina viene aggiunto un div che contiene il titolo (della sala), un'immagine e una breve descrizione. Questo div viene realizzato così:

```
<div class="blog-post-thumb">
  <div class="blog-post-image" id="[num]">
    
  </div>
  <div class="blog-post-title">
    <h3>[Titolo]</h3>
  </div>
  <div class="blog-post-des">
    <p>[Descrizione]</p>
  </div>
</div>
```

Problemi riscontrati e soluzioni adottate

Oggi Matan e Mattia hanno avuto un problema di comprensione dei requisiti. Infatti Mattia ritiene che le promozioni devono essere mostrate a tutti gli utenti con titolo, immagine e descrizione, mentre Matan pensa che esse debbano essere composte unicamente dal messaggio e mostrate solo ad alcuni utenti. È stata mandata un'email al cliente per chiarimenti a riguardo e abbiamo intenzione di parlarne al prossimo stand-up.

Punto della situazione rispetto alla pianificazione

Rispetto alla pianificazione siamo in orario.

Programma di massima per la prossima giornata di lavoro

Per la prossima lezione l'obiettivo di Thor e Forni sarà far funzionare i test sul server di prova e successivamente sul server originale.