Diario di lavoro

|  |  |
| --- | --- |
| Luogo | Scuola arti e mestieri Trevano |
| Data | 12.11.2019 |

|  |
| --- |
| Lavori svolti |
| Oggi mi sono dedicato a finire la modifica sia dei programmi sia dei database.  Per iniziare ho finito quella dei programmi che consisteva semplicemente nel aggiungere una parte del backend semplice molto simile a quella per la modifica degli utenti.  */\*\*  \* Questo metodo modifica le impostazioni di un programma  \* @param $id Int Id del programma da modificare  \*/* public function updateSettings($id){  if(validate::*checkInt*($id)){   */\*\*  \* Metto in sicurezza il nome prima di inserirlo nella query  \*/* $name = validate::*secureString*($\_POST['smName']);  if ($name != NULL) {  db\_manager::*execute*(  "UPDATE settings set name = '" . $name . "' where id = " . $id  );  }   */\*\*  \* Controllo se è stato selezionato almeno un giorno  \*/* $days = $this->getDays();  if($days != ''){  db\_manager::*execute*(  "UPDATE settings set day = '" . $days . "' where id = " . $id  );  }   */\*\*  \* Trasformo l'ora da una stringa ad un int e controllo che sia effettivamente un intero  \*/* $hour = (int)*substr*($\_POST['smhour'], 0, 2);  if (validate::*checkInt*($hour)) {  db\_manager::*execute*(  "UPDATE settings set hour = '" . $hour . "' where id = " . $id  );  }   */\*\*  \* Controllo che i backup da tenere sia almeno 1 e che sia un numero.  \*/* if (validate::*checkInt*($\_POST['smda']) && $\_POST['smda'] >= 1) {  db\_manager::*execute*(  "UPDATE settings set delete\_after = '" . $\_POST['smda'] . "' where id = " . $id  );  }  }   */\*\*  \* Ritorno alla pagina dei database  \*/* $this->home(); }    Questo metodo è suddiviso nei diversi input a cui vengono fatti i controlli e di cui poi viene immesso il valore all’interno del database.  In seguito ho fatto sia il frontend che il backend della modifica dei database a cui per motivi di sicurezza non si può modificare l’host il nome utente o la password.  Nel frontend come si vede si può solo modificare il programma assegnato al database.  */\*\*  \* Questa funzione cambia il programma di un database  \* @param $id Int Id del database da modificare  \*/* public function modify($id){   */\*\*  \* Controllo che è stato passato è un id  \*/* if(validate::*checkInt*($id)) {   */\*\*  \* Controllo che l'impostazione che è stata passata è un id  \*/* if (validate::*checkInt*($\_POST['dmSetting'])) {   */\*\*  \* Controllo che ci sia un programma con quel id  \*/* $results = db\_manager::*executeQuery*("SELECT *\** from settings where id ='" . $\_POST['dmSetting'] . "'");  if($results !== NULL) {    db\_manager::*execute*(  "UPDATE database\_data set settings\_id = '" . $\_POST['dmSetting'] . "' where id = " . $id  );    }  }  }   $this->home(); }  La funzione come quella di prima è molto semplice e in questa visto che c’è solo un input è semplicemente un’azione sul database, che viene eseguita dopo aver controllato che il valore passato sia valido e che sia un programma esistente. |

|  |
| --- |
| Problemi riscontrati e soluzioni adottate |
| - |

|  |
| --- |
| Punto della situazione rispetto alla pianificazione |
| Sono in linea con la pianificazione. |

|  |
| --- |
| Programma di massima per la prossima giornata di lavoro |
| Mettere a posto il Gantt e il documento che spiega il mio problema. |