Diario di lavoro

|  |  |
| --- | --- |
| Luogo | Arti Mestieri Trevano |
| Data | 08.11.2019 |

|  |
| --- |
| Lavori svolti |
| Ho modificato la validazione del nome della vasca:  public function validateTankName($tankName) {  $validElement = $this->generalValidation($tankName);  $pattern = '/^[A-Za-z0-9\_-]\*$/';  $arrayAllNameTank = $this->multidimensionalArrayToNormalArray($this->allNameTank);   if (*preg\_match*($pattern, $validElement) && *strlen*($validElement) > 0 && *strlen*($validElement) <= 45) {  if (*in\_array*($tankName, $arrayAllNameTank)){  return self::*ALREDY\_EXIST*;  }else{  return self::*CORRECT\_NAME*;  }  }else{  return self::*INCORRECT\_NAME*;  } }  Nel tankValidation:  if($this->validationFunction->validateTankName($tank["tankName"]) == ValidationFunction::*ALREDY\_EXIST*){  $this->stringErrors = "Il nome della vasca esiste già";  $validationOK = false; }else if($this->validationFunction->validateTankName($tank["tankName"]) == ValidationFunction::*INCORRECT\_NAME*){  $this->stringErrors = "Il nome della vasca è sbagliato";  $validationOK = false; }  Ho modificato anche le funzioni della modifica per far si che la validazione funzioni anche quando si modificano i dati degli acquari.  public function requirePageModifyForm($name){  $title = "Modifica vasca";  $nameButton = "Modifica";  $path = *URL* . "tankManagement/modifyTank/".$name;  $stringErrors = $this->tankValidatioModel->stringErrors;   if($this->arrayTank!=null){  $tankName = $this->arrayTank["tankName"];  $calcium = $this->arrayTank["calcium"];  $magnesium = $this->arrayTank["magnesium"];  $kh = $this->arrayTank["kh"];  $waterChange = $this->arrayTank["waterChange"];  $liter = $this->arrayTank["liter"];  }else{  $tankToModify = $this->tankManagementModel->getByName($name);  $tankName = $tankToModify[0]["nome"];  $magnesium = $tankToModify[0]["calcio"];  $calcium = $tankToModify[0]["magnesio"];  $kh = $tankToModify[0]["kh"];  $waterChange = $tankToModify[0]["ultimo\_cambio\_acqua"];  $liter = $tankToModify[0]["Litri"];  }   require "GestioneAcquariMarini/views/\_templates/header.php";  require "GestioneAcquariMarini/views/\_templates/menu.php";  require "GestioneAcquariMarini/views/gestioneAcquari/tank/tankManagement.php";  require "GestioneAcquariMarini/views/\_templates/footer.php"; }  public function formModifyTank($name) {  *session\_start*();  if ($\_SESSION["authentification"] == true) {  $this->requirePageModifyForm($name);  } else {  *header*("Location:" . *URL*);  } }  Poi ho inziato a fare la stessa cosa per la convalidazione dell’utente nello stesso modo. |

|  |
| --- |
| Problemi riscontrati e soluzioni adottate |
| Oggi mi sono bloccato mentre stavo Sistema la validazione del nome della vasca infatti nel controllare se il nome ce già il nome usavo la seguente funzione:  if (*in\_array*($tankName, $arrayAllNameTank)){  return self::*ALREDY\_EXIST*; }else{  return self::*CORRECT\_NAME*; }  E la funzione in\_array non funzionava e non capivo il motivo, così ho perso un sacco di tempo a capire il motivo per cui non funzionava utilizzando anche array\_search e testando il più possibile tutti i valori che inserivo, alla fine ho capito che tutte le funzioni di php che ti permettono di capire se un valore è già presente nell’array non accettono array multidimensionali, per risolvere il problema ho fatto questa funzione che converte l’array multidimensionale in un normale array.  private function multidimensionalArrayToNormalArray($arrayMultidimensional){  $result = array();  foreach ($arrayMultidimensional as $key => $value) {  $result[] = $value["nome"];  }  return $result; } |

|  |
| --- |
| Punto della situazione rispetto alla pianificazione |
| In anticipo di una settimana |

|  |
| --- |
| Programma di massima per la prossima giornata di lavoro |
| -Finire la convalidazione lato client e server per gli utenti |