Diario di lavoro

|  |  |
| --- | --- |
| Luogo | Arti Mestieri Trevano |
| Data | 03.12.2019 |

|  |
| --- |
| Lavori svolti |
| Oggi ho finito l’implementazione della gestione degli abitanti, nel model ho fatto i metodi per aggiungere e modifcare i dati nel database:  public function add($habitant) {  $addHabitant = "INSERT INTO abitante (specie,genere,tipo,numero,nome\_vasca) VALUES (?,?,?,?,?)";  $this->statement = $this->connection->prepare($addHabitant);  $this->statement->bindParam(1, $habitant["species"]);  $this->statement->bindParam(2, $habitant["sex"]);  $this->statement->bindParam(3, $habitant["type"]);  $this->statement->bindParam(4, $habitant["habitantNumber"]);  $this->statement->bindParam(5, $\_SESSION["referencesTankName"]);  $this->statement->execute(); } public function modify($habitant, $species, $sex) {  $modifyHabitant = "UPDATE abitante SET specie=:species, genere=:sex,tipo=:type,numero=:habitantNumber WHERE specie=:orginalSpecies && genere=:originalSex && nome\_vasca=:tankName";  $this->statement = $this->connection->prepare($modifyHabitant);  $this->statement->bindParam(':orginalSpecies', $species, PDO::*PARAM\_STR*);  $this->statement->bindParam(':originalSex', $sex, PDO::*PARAM\_STR*);  $this->statement->bindParam(':species', $habitant['species'], PDO::*PARAM\_STR*);  $this->statement->bindParam(':sex', $habitant['sex'], PDO::*PARAM\_STR*);  $this->statement->bindParam(':type', $habitant['type'], PDO::*PARAM\_STR*);  $this->statement->bindParam(':habitantNumber', $habitant['habitantNumber'], PDO::*PARAM\_STR*);  $this->statement->bindParam(':tankName', $\_SESSION["referencesTankName"], PDO::*PARAM\_STR*);  $this->statement->execute(); }  Poi nel controller ho gestito i vari bottoni e le varie azioni che bisogna eseguire, per l’aggiunta ho usato semplicemente questo metodo:  public function addHabitant(){  $this->habitantArrayManagement = $this->getArrayHabitantByPost();  $habitant = $this->habitantArrayManagement;  if($this->habitantValidation->validation($habitant, null, null)){  $this->habitantModel->add($habitant);  $this->habitantArrayManagement = null;  }  $this->stringErrors = $this->habitantValidation->stringErrors;  $this->showAllHabitantsTank($\_SESSION["referencesTankName"]); }  Mentre per la modifca una funzione che mi riporta i dati da modificare nel form e un che mi faccia la modifica:  public function modifyHabitant($species, $sex){  $habitant = $this->habitantModel->getAllHabitantBySpeciesAndSex($species,$sex);  $this->habitantArrayManagement = $this->getArrayHabitantByDatabase($habitant);  $this->textButtonForm="Modifica";  $this->pathForm=*URL* . "habitantManagement/updateHabitant/".$species."/".$sex;  $this->showAllHabitantsTank($\_SESSION["referencesTankName"]); }  public function updateHabitant($species, $sex){  $this->habitantArrayManagement = $this->getArrayHabitantByPost();  $habitant = $this->habitantArrayManagement;  $habitantByDatabase = $this->getArrayHabitantByDatabase($this->habitantModel->getAllHabitantBySpeciesAndSex($species,$sex));  if($this->habitantValidation->validation($habitant, $habitantByDatabase["species"], $habitantByDatabase["sex"])){  $this->habitantModel->modify($habitant, $habitantByDatabase["species"], $habitantByDatabase["sex"]);  $this->habitantArrayManagement = null;  $this->textButtonForm="Aggiungi";  $this->pathForm=*URL* . "habitantManagement/addHabitant";  }else{  $this->textButtonForm="Modifica";  $this->pathForm=*URL* . "habitantManagement/updateHabitant/".$species."/".$sex;  }  $this->stringErrors = $this->habitantValidation->stringErrors;  $this->showAllHabitantsTank($\_SESSION["referencesTankName"]); }  Per cancellare un vasca prima ho dovuto cancellare tutti gli abitanti correlati alla vasca:  public function delete($name){  $deleteHabitantTank = "DELETE FROM abitante WHERE nome\_vasca=:nameTank";  $this->statement = $this->connection->prepare($deleteHabitantTank);  $this->statement->bindParam(':nameTank', $name , PDO::*PARAM\_STR*);  $this->statement->execute();  $deleteTank = "DELETE FROM vasca WHERE Nome=:nameTank";  $this->statement = $this->connection->prepare($deleteTank);  $this->statement->bindParam(':nameTank', $name , PDO::*PARAM\_STR*);  $this->statement->execute(); }  Questi due metodi li uso per avere un array dell’abitante, nel primo caso dal database e nel secondo dal post del form:  public function getArrayHabitantByPost(){  $species = $\_POST["species"];  $sex = $\_POST["sex"];  $type = $\_POST["type"];  $habitantNumber = $\_POST["habitantNumber"];  $habitant = array("species"=>$species,"sex"=>$sex,"type"=>$type,"habitantNumber"=>$habitantNumber);  return $habitant; }  public function getArrayHabitantByDatabase($habitant){  $species = $habitant[0]["specie"];  $sex = $habitant[0]["genere"];  $type = $habitant[0]["tipo"];  $habitantNumber = $habitant[0]["numero"];  $habitant = array("species"=>$species,"sex"=>$sex,"type"=>$type,"habitantNumber"=>$habitantNumber);  return $habitant; }  Poi ho fatto alcuni test personalmente e ho fatto fare anche un test del prodotto a Carlo.  Ho sistemato alcune cose come il fatto che per gli utenti normali non esce neanche nel menu la possibilità di andare in gestione utenti. |

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

|  |
| --- |
| Punto della situazione rispetto alla pianificazione |
| In ritardo di 8 ore. |

|  |
| --- |
| Programma di massima per la prossima giornata di lavoro |
| -Documentare il tut |