MINISTERUL EDUCAȚIEI NAȚIONALE

UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ŞI CALCULATOARE
DEPARTAMENTUL AUTOMATICA

Electronics Inventory

Autor: **Darius Ionica**

1. Requirements

The project aims to simplify the management of someone's electronics inventory. The user, as long as he/she has access, is able to add any of the specified types of electronic components to a database, delete them, update them and see the full inventory of the specified item.

2. Presentation of the proposed solutions

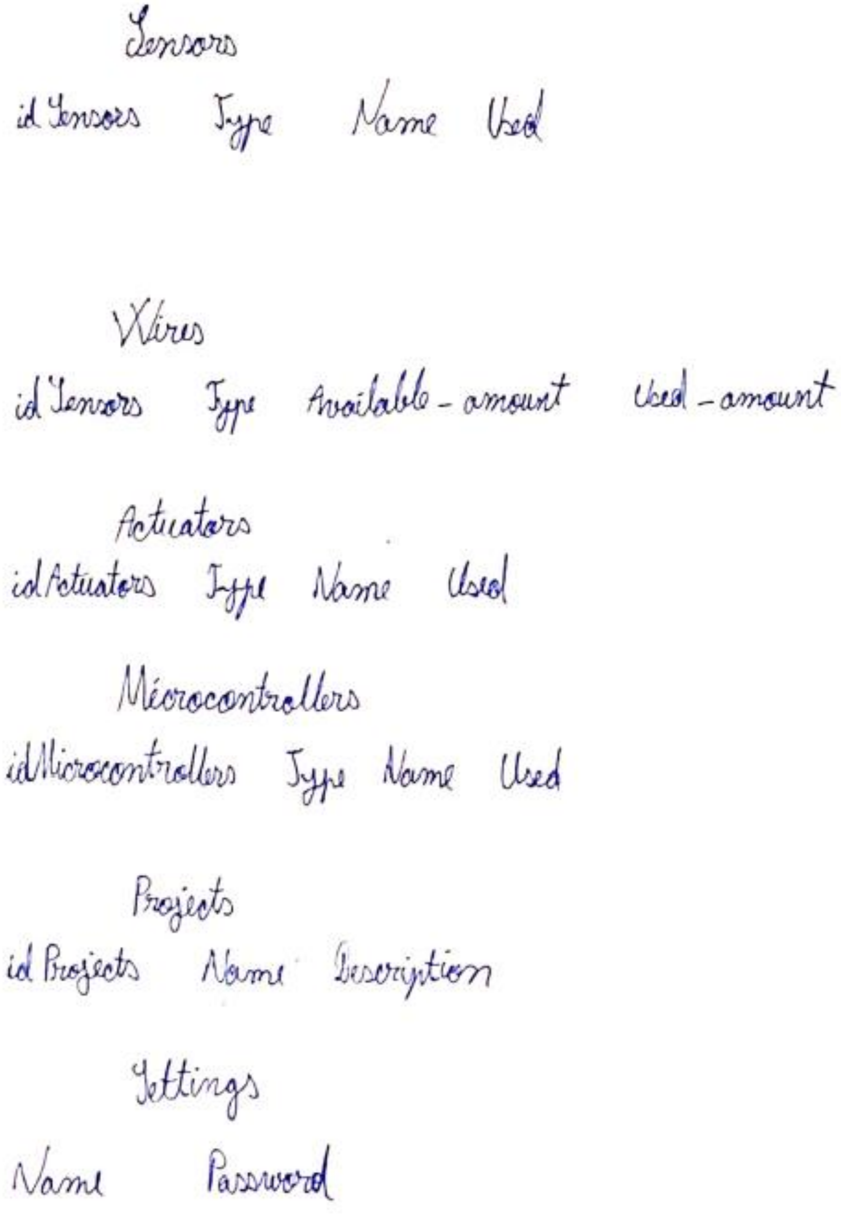
From a more theoretical point of view, the project will be able to display the inventory of: sensors, wires, actuators, microcontrollers, and the projects in which all the early mentioned electronics are used. The access to editing the inventory is done by logging into an account. Once you are logged into your account, if you have permision to update the inventory you will be able to do that, otherwise a message will be displayed. There is also a settings part on the main page from where you can add a new user, delete an existing one and update one.

3. Justification of the chosen topic

I chose this kind of project in order to reduce the time when I am working on a project. It happened very often to me to search for a certain sensor that I knew I had and I just couldn't find it. Since I can write all the owned electronics in this database, it will give me a brief idea whether the electronic which I'm looking for is in use or not.

4. Phases of the project's developement

The project started out by writing on a paper the desired table names and the name of the collums which will be used further in the project. (Figure 1)



Sensors

id Sensors Type Name Used

Wires

id Sensors Type Available-amount Used-amount

Actuators

idActuators Type Name Used

Microcontrollers

idMicrocontrollers Type Name Used

Projects

id Projects Name Description

Settings

Name Password

Figure 1. The sketch of the tables

After all the names of the collums were inserted and the tables with the specific names were created, I went on to code the web interface of the project, starting with the login part for the user.

The login part is related to one of the 6 tables created in MySql Workbench. There you can login with diffeent existing users. The different users have different type of access to the available operations. (Figure 2)

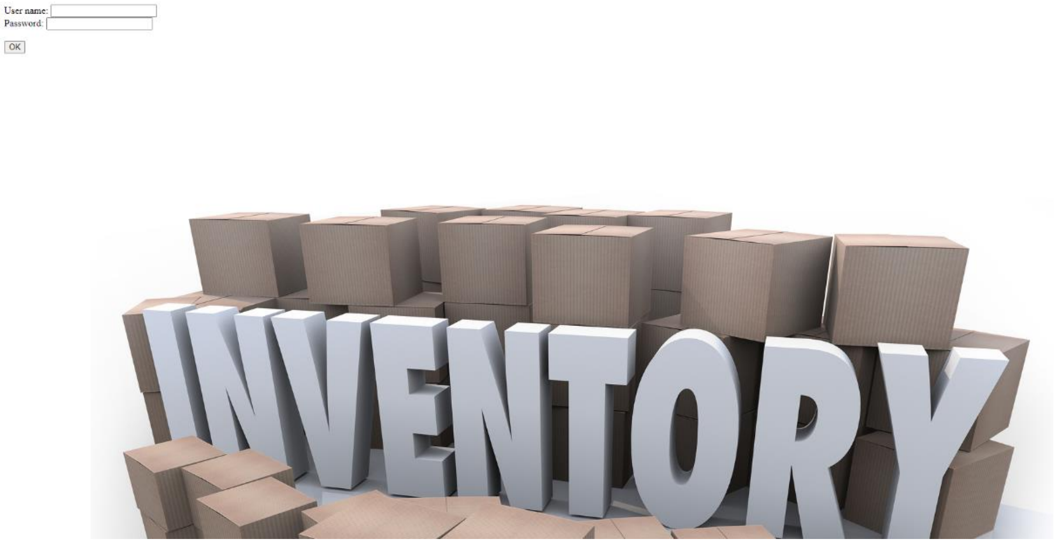
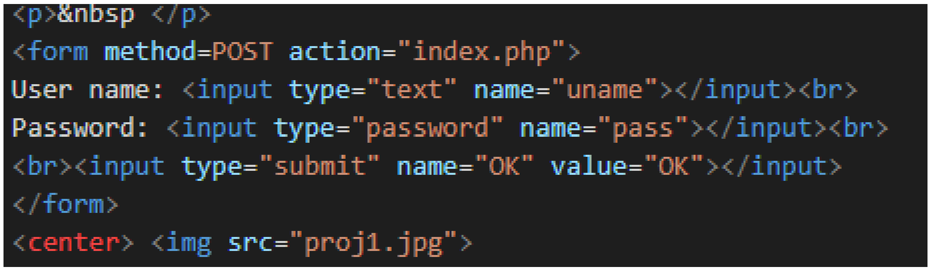


Figure 2. Login screen

This is the first thing you see once you enter the site. After you succesfully login you will be taken to the main page.

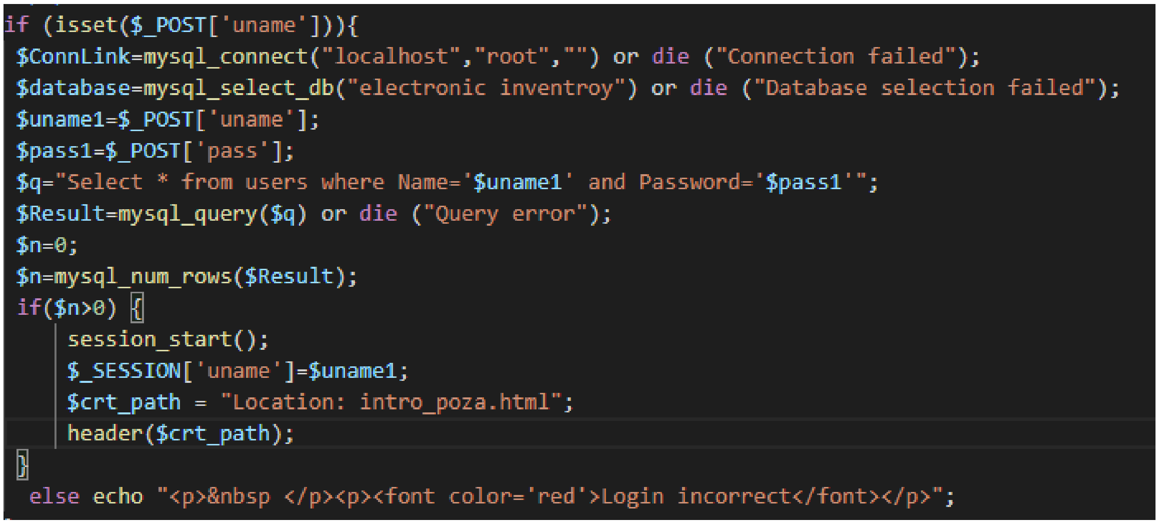
Next up, I will present to you the code for this part. It can be found in the ”index.php” file. (Figure 3)



```
<p>&nbsp </p>
<form method=POST action="index.php">
User name: <input type="text" name="uname"></input><br>
Password: <input type="password" name="pass"></input><br>
<br><input type="submit" name="OK" value="OK"></input>
</form>
<center> <img src="proj1.jpg">
```

Figure 3. Buttons and text boxes

This part just creates the text boxes and the submit button from the above picture. It also adds the inventory picture below them. (Figure 4)



```
if (isset($_POST['uname'])){
 $ConnLink=mysql_connect("localhost","root","") or die ("Connection failed");
 $database=mysql_select_db("electronic inventroy") or die ("Database selection failed");
 $uname1=$_POST['uname'];
 $pass1=$_POST['pass'];
 $q="Select * from users where Name='$uname1' and Password='$pass1'";
 $Result=mysql_query($q) or die ("Query error");
 $n=0;
 $n=mysql_num_rows($Result);
 if($n>0) {
     session_start();
     $_SESSION['uname']=$uname1;
     $crt_path = "Location: intro_poza.html";
     header($crt_path);
 }
  else echo "<p>&nbsp </p><p><font color='red'>Login incorrect</font></p>";
```

Figure 4. Creation of the session varaible and database connection

The above part connects to the database and selects every user and password from the specified table. Creating a session variable for the username helped me further in the project when I wanted to restrict the access to normal users. If the inserted username and password combination doesn't match any from the users table, the access to the operation will be totally restricted with a red coloured message.

Next up, is the main page of the project. From here you can go on to do different operations on the contents of each table. (Figure 5)

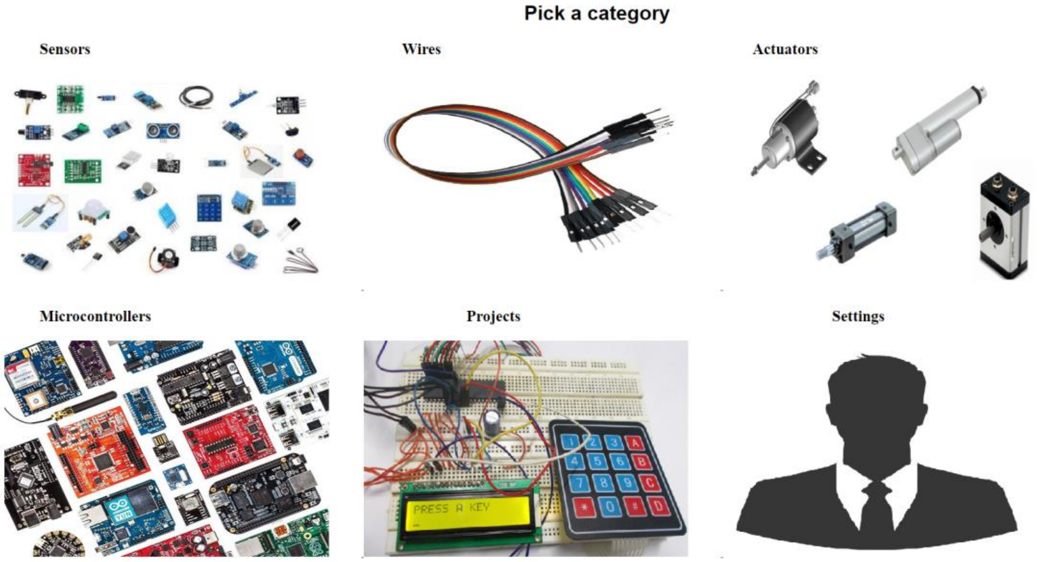
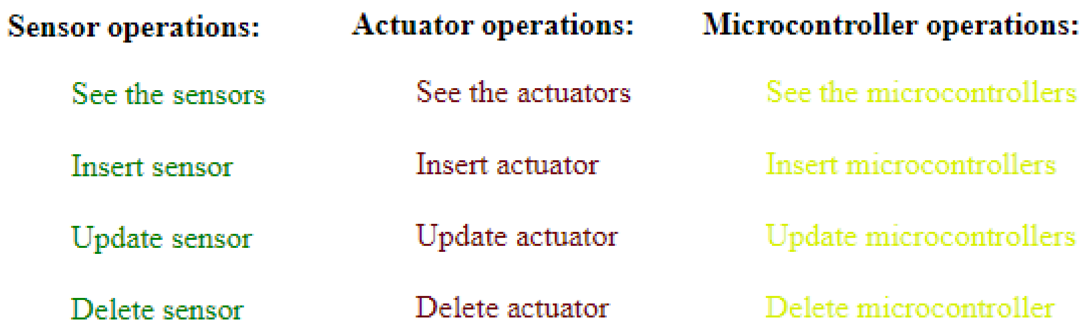


Figure 5. Main page

Since the Sensors, Actuators and Microcontrollers are very much alike, I will present only one operation for each. (Figure 6)

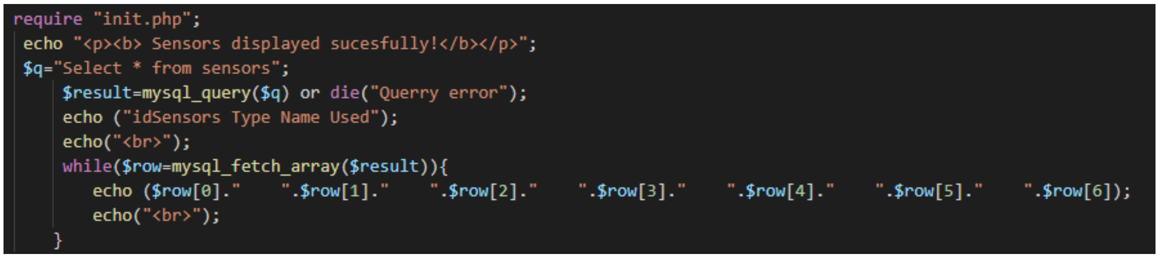


Sensor operations:	Actuator operations:	Microcontroller operations:
See the sensors	See the actuators	See the microcontrollers
Insert sensor	Insert actuator	Insert microcontrollers
Update sensor	Update actuator	Update microcontrollers
Delete sensor	Delete actuator	Delete microcontroller

Figure 6. Available operations

Once you click the Sensors image, you will be redirected to a page from where you can do the operations on the specific table. „See the sensors” will show you every sensor from the table.

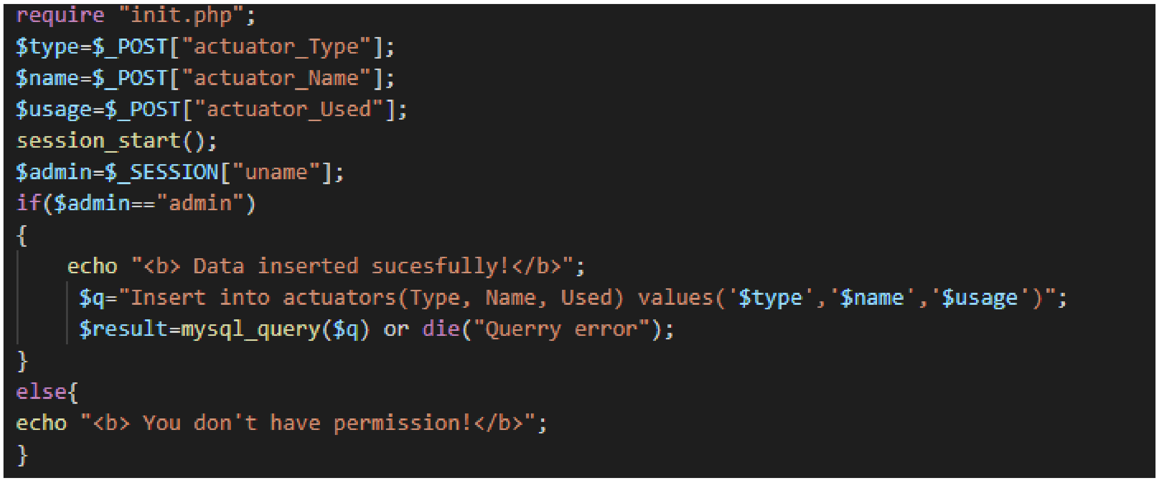
The code for this part will simply select every content of the sensors table and display it for the user. This operation can be done by any logged user. (Figure 7)



```
require "init.php";
 echo "<p><b> Sensors displayed sucesfully!</b></p>";
 $q="Select * from sensors";
     $result=mysql_query($q) or die("Querry error");
     echo ("idSensors Type Name Used");
     echo("<br>");
     while($row=mysql_fetch_array($result)){
        echo ($row[0]."    ".$row[1]."    ".$row[2]."    ".$row[3]."    ".$row[4]."    ".$row[5]."    ".$row[6]);
        echo("<br>");
    }
```

Figure 7. Printing the contents of the tables

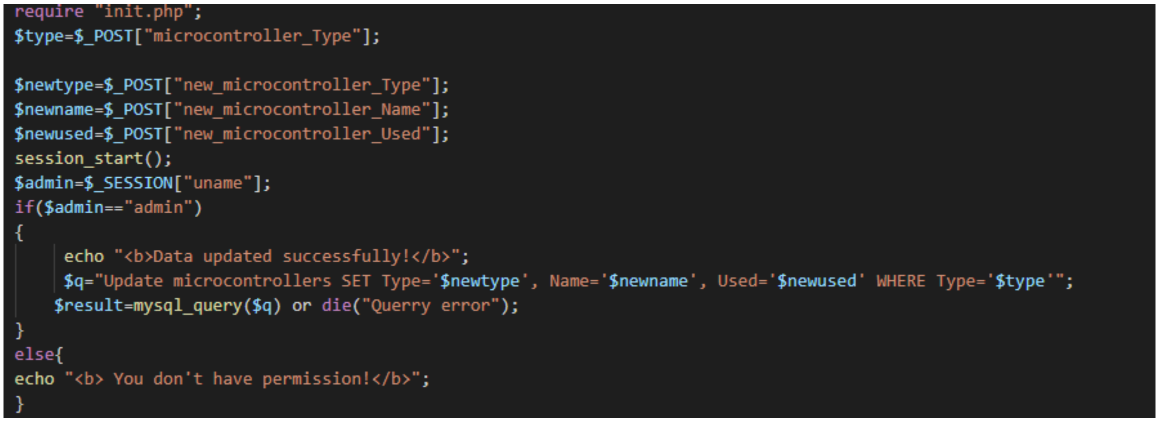
The „Insert actuator” link will enable the user to insert an actuator/sensor/microcontroller to their own tables. You will be required to write the Type, Name and Used status for any of the three. These are the parts where only the username containing „admin” has access. (Figure 8)



```
require "init.php";
$type=$_POST["actuator_Type"];
$name=$_POST["actuator_Name"];
$usage=$_POST["actuator_Used"];
session_start();
$admin=$_SESSION["uname"];
if($admin=="admin")
{
    echo "<b> Data inserted sucesfully!</b>";
     $q="Insert into actuators(Type, Name, Used) values('$type','$name','$usage')";
     $result=mysql_query($q) or die("Querry error");
}
else{
echo "<b> You don't have permission!</b>";
}
```

Figure 8. Inserting new data in the tables

The „update microcontroller” link will let the user, as long as the username is admin, to edit the Name, Type and Used status for any of the existing microcontrollers/sensor/actuators from their tables. The code for this part: (Figure 9)



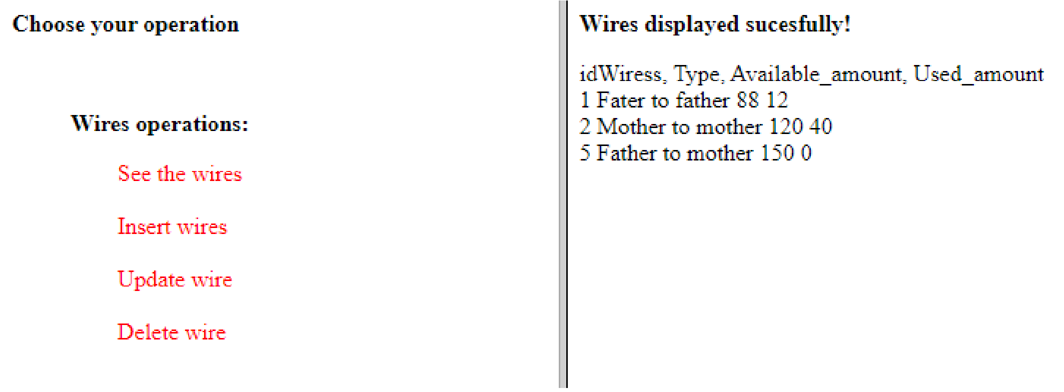
```
require "init.php";
$type=$_POST["microcontroller_Type"];

$newtype=$_POST["new_microcontroller_Type"];
$newname=$_POST["new_microcontroller_Name"];
$newused=$_POST["new_microcontroller_Used"];
session_start();
$admin=$_SESSION["uname"];
if($admin=="admin")
{
     echo "<b>Data updated successfully!</b>";
     $q="Update microcontrollers SET Type='$newtype', Name='$newname', Used='$newused' WHERE Type='$type'";
    $result=mysql_query($q) or die("Querry error");
}
else{
echo "<b> You don't have permission!</b>";
}
```

Figure 9. Updating data from tables

Lastly, you can delete any of the three by clicking the specified link. This link will require the user to specify the Type, Name and Used status. If the Used status is „Yes” the operation will stop, since that means the sensor for example is used in a project, and a message will appear.

Moving on to the Wires part. Since there are mainly only three types of wires used in projects, I restricted the Type of wire for any operation. As I said before, you can See the wires from the table, Insert, Update, or Delete one type. The user will be required to specify the available and used amount for any type of the wire which will be inserted. If there are any wires from that category in use, a message will be displayed and the action will stop, in the case that the user wants to delete one type. The coding for this part pretty much resembles the ones mentioned before, so you can find them in the .rar file containing the „Wires” word. (Figure 10)

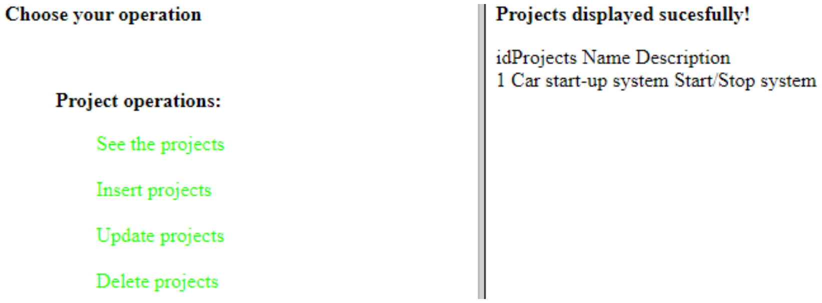


Wires displayed sucesfully!

```
idWiress, Type, Available_amount, Used_amount
1 Fater to father 88 12
2 Mother to mother 120 40
5 Father to mother 150 0
```

Figure 10. Wires operations

Next up is the Projects part. Here the user will be able to see the Name of the project, and a brief description about it. As before, all the earlier mentioned operations will be available here, of course with the „admin” username restriction. The code resambles the above described one, so it can be found in more details in .rar file with the files containing Project in their name. (Project 11)



```
Projects displayed sucesfully!

idProjects Name Description
1 Car start-up system Start/Stop system
```

Figure 11. Projects operations

The Settings picture will enable the user to Insert, Update or Delete an existing user. These operations will only be available to the user with the username „admin”, just as the other similar operations. In order to give access to a newly inserted user, You will have to pick for the username the word „admin”, the password can be as you wish. The code for this part can be found in the .rar file in the files containing the „User” word. (Figure 12)

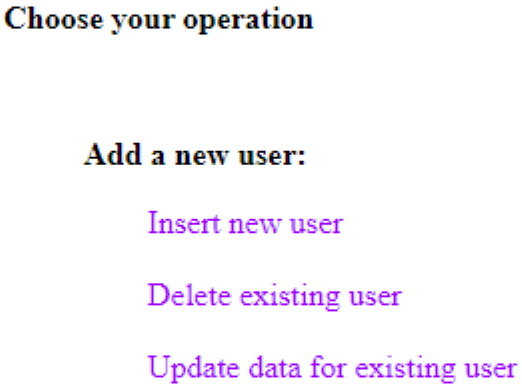


Figure 12. Settings operations

5. Results

In developing this project, I’ve coded in .html to create the web interface, in .php to create the relations between the internet pages and the tables from the data base, in .css to add a colorful touch to the clickable links, and in MySql Workbench in order to create the database and the tables containing the electronic devices.

In the end, this will enable me, and any other user of the website, to keep the electronics inventory up to date, and provide an easier way to find the electronic part you are looking for, and know if it is already in use or not. A further upgrade to the project is to add more types of electronics, creating both operation and tables for them.