

CODE

HTML SECTION

```
<!DOCTYPE html>

<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Moisture calculator</title>
  <link rel="image/jpg" href="indexx.png">
  <link rel="stylesheet" href="css.css">
</head>
<body>
  <div id="soil">
    <form id=form1>
      <h1>Moisture Determination System</h1> <hr>

      <div>
        <label >Temperature: </label>
        <input type="number" id="temp" placeholder="input temperature" min="1">
      </div><br>
      <div>
        <label >Dry weight: </label>
        <input name ="dry" type="number" id="dry" placeholder="input sampled dry weight" min="1">
      </div> <br>
      <div>
        <label >wet weight: </label>
        <input type="number" id="wet" placeholder="input sampled wet weight" min="1">
      </div><br>
    </form>
  </div>
</body>
</html>
```

```

    <div>
      <label>water level</label>
      <input type="range" max="100" min="0" value="10" id="level">
    </div> <br>
    <hr>
    <div>
      <label>moisture:</label> <span class="moisture">0</span>
    </div>
  </form>

</div> <script src="assignment.js"></script>
</body>
</html>

```

CSS STYLING SECTION

```

body{
  font-size: 1.2em;
  width:100%;
  margin: 0 auto;
  letter-spacing: 0.5px;
}
h1 {
  text-align: center;
}
#soil{

```

```
padding: 1.5em;
background: linear-gradient(-45deg, rgb(245,316,127), rgb(241,81,38));
border-radius: 10px;
}
div{
  line-height: 2.5em;
}
input[type="number"]{
  font-size: 1.1em;
  text-align: center;
}
label, input{
  width: 100%;
}
input[type="submit"]{
  width: auto;
}
@media(min-width:700px){
  body{
    width: 50%;
    background:url(indexx.png) no-repeat center center fixed;
    background-size: cover;
    margin-top: 20px;
  }
  #soil{
    opacity: 0.9;
  }

  label, input{
    display: inline-block;
```

```
    width: 40%;  
  }  
  div {  
    line-height: 2.7em;  
  }  
}
```

JAVASCRIPT SECTION

```
do {  
    password=prompt("input password","password")  
} while (password !="shoemaker");  
var temp = prompt("input temperature");  
var wetWeight= prompt("wet weight of soil");  
var dryWeight = prompt("dry weight of soil")  
var level = document.getElementById('level')  
  
document.onload(function(){  
    var moisture = (wetWeight/dryWeight)*100;  
    document.getElementById('moisture').innerHTML = moisture  
})  
document.getElementById("temp").innerHTML= temp  
document.getElementById("wet").innerHTML= wetWeight  
document.getElementById("dry").innerHTML= dryWeight  
if (level<3){  
    alert('add water to tank')  
}  
else if (temp>100 ||moisture<=70){
```

```

var showalert= true
setInterval(function(){
    showalert=false
}, 60000)
function jsAlert(){
    if (showalert){
        alert("turn on water system")
    }
}
else {return 0}

```

USING THE SOFTWARE DEVELOPMENT CYCLE A PROGRAM WAS CREATED

CONCEPTUALIZATION: the project is to create a system that reads the temperature of the soil, determines the moisture content of the soil, configure the time interval for a water release in the system, and to trigger an alarm when there is need for water

SPECIFICATIONS: The project will need a GUI(graphical user interface) which will be used in the writing and construction of the program; firm knowledge on plant and soil structures; a storage database; automated controllers and a few testers and transducers

DESIGN: The design of the system will take the use of programming and scripting languages such as HTML, CSS e.t.c for the front end design and structure of the program while JavaScript, python, SQL and PHP can be used for back end functionality of the program

IMPLEMENTATION:

Steps to Implementation

- *The structure of the interface will be developed*
- *The design layout of the application will be done with css*
- *The functionality and back end structure of the program will be made with JavaScript or python*
- *The data will be received stored and relayed to machined remotely to perform their tasks*

TESTING AND DEBUGGING

The program will be tested as its being developed. official testing of the program will be done on 14th of may 2020

RELEASE OF SOFTWARE

The software release date is on the 17th of may 2020. Although this can be pushed forward if the deadline is not met.

HARDWARE FEATURS

The hardware components of the system include thermometers for soil temperature measurements, a scale for measurement of soil samples, transducer for converting heat energy to digital signals, actuators for controlling the physical aspects of the project, such as the releasing of water, when needed by the automated system. A physical server that will relay information either wirelessly or through Ethernet cables.

SOFTWARE FEATURES

The need of a GUI is crucial for the project, a hosted server where the program will run and retrieve the necessary files and programs.

ALGORITHM

1. Start

2. Read password

3. Password= "document"

4. do

- Input password

While password not equals "document"

- Open new page

5. Read temperature, moisture, water, level

6. If level ≤ 2

- Alert " WARNING, WATER LEVEL LOW, PLEASE REFILL THE TANK"

Else If moisture ≤ 70 and water ≤ 80

- Then water equal true

Else

- Water equals false

7. For water equals true

Turn on water system for 60 seconds

8. End

FLOWCHART



