



Sistem Automatizat de Control al Irigațiilor în Agricultură

## iCASA

Irrigation Control Automated System for Agriculture

### Secțiune

B . Științe aplicate include proiecte din domeniile: inginerie mecanică, energetică și termotehnică, electrotehnică și electronică, echipamente și tehnologii de măsurare sau monitorizare, sisteme automate, sisteme autonome sau controlabile de la distanță ce îmbină parte mecanică și/sau electrică și/sau electronică, reciclare, economie etc.

### Categorie

Juniori

### Scop

Realizarea unui sistem automatizat de control al sistemelor de irigații în agricultură prin îmbinarea soluțiilor IOT și modelelor predictive în meteorologie

### Obiective

Construirea unui sistem electronic de:

- consolidare a datelor colectate de senzori
- interpretarea datelor în timp real
- stocarea datelor istorice
- comunicarea sistemului de control cu platforme mobile pentru control direct
- controlul sistemelor de management al resurselor

### Problema identificată spre rezolvare

În contextul schimbărilor climatice, când resursele acvifere devin tot mai scăzute, este foarte importantă folosirea rațională a apei și controlul precis al cantităților folosite în agricultură.

Pentru a reduce cât mai mult risipa apei, este esențială optimizarea mai multor aspecte:

1. Controlul asupra condițiilor atmosferice naturale, prin măsurători precise și istorice ale variabilelor temperatură atmosferică, umiditate atmosferică, umiditate a solului.
2. Controlul asupra cantității de apă folosită pentru irigare, în funcție de: tipul de cultură agricolă, ținând seama de caracteristicile fiecărei plante; menținerea în parametrii optimi ai variabilelor măsurate; gestiunea rezervelor de apă pe baza datelor istorice colectate și a previziunilor algoritmice.

<b>Echipa de proiect</b>	Gala Călușer – Clasa a V - a B Colegiul Național Pedagogic "Regina Maria", Deva, jud. Hunedoara <i>Rol: Construirea prototipului și codul Arduino</i>
	David Trif – Clasa a V - a B Colegiul Național Pedagogic "Regina Maria", Deva, jud. Hunedoara <i>Rol: Design funcțional și schema prototipului</i>
<b>Colaboratori Externi</b>	Echipa de Robotică a Centrului de Excelență Hunedoara, jud. Hunedoara Coordonator: Profesor Paul Țoța
<b>Coordonatorul Echipei</b>	Teodora Scânteie – Profesor Informatică Colegiul Național Pedagogic "Regina Maria", Deva, jud. Hunedoara
<b>Etape Parcurse</b>	<ol style="list-style-type: none"> <li>1. <b>Designul schematic al iCASA</b> – conceptul schematic al tuturor componentelor electronice ce vor constitui sistemul automat de control al irigațiilor</li> <li>2. <b>Montajul prototipului</b> folosind ca placă principală de dezvoltare Elegoo Uno și un sistem de vizualizare a datelor folosind ecran LCD 16X2</li> <li>3. <b>Conectarea senzorilor</b> pentru colectarea datelor</li> <li>4. <b>Conectarea modului Bluetooth</b> pentru comunicare cu platforma mobile</li> <li>5. <b>Crearea codului pentru programarea prototipului</b>, folosind Arduino IDE</li> <li>6. <b>(În lucru) Crearea aplicației mobile OPERA</b> (Optimizare Predictivă a Economiei Resurselor de Apă) folosind MIT App Inventor</li> </ol> <p>Aplicația OPERA (Optimizare Predictivă a Economiei Resurselor de Apă) propusă spre a fi dezvoltată are rolul unei stații meteo pentru monitorizare climatică.</p> <p>Aplicația va:</p> <ul style="list-style-type: none"> <li>- prelua informațiile (temperatură, umiditate aer-sol) de la senzori</li> <li>- va afișa în timp real datele colectate și le va stoca pentru a permite analize ulterioare</li> <li>- va oferi grafice zilnice și lunare pentru parametrii meteorologici măsurați</li> <li>- va utiliza algoritmi pentru prognoza meteo</li> </ul> <ol style="list-style-type: none"> <li>7. <b>(În lucru) Conectarea sistemului de control la releele sistemului de irigații</b> dezvoltat de echipa colaboratoare.</li> </ol>

<b>Metode Folosite</b>	<p>Prototiparea – Încercări empirice repetate folosind</p> <ul style="list-style-type: none"> <li>a) Placa de dezvoltare Elegoo Uno</li> <li>b) Senzor de temperatură și umiditate DHT11</li> <li>c) Modul Bluetooth HC05</li> <li>d) Ecran LCD (Liquid Crystal Display) 16X2</li> <li>e) Potențiometru 10 kohmi</li> <li>f) Baterie 9 V pentru alimentarea sistemului</li> <li>g) Cablu de conectare Arduino cu doi pini (jump wires)</li> <li>h) Senzor de umiditate a solului HW-080</li> <li>i) Scut Ethernet pentru conectarea la Internet prin LAN și stocarea datelor în cardul MicroSD</li> <li>j) Scut pentru Senzori V 5.0 pentru consolidarea conexiunilor și optimizarea folosirii cablajului</li> <li>k) Releu dublu pentru circuite de tensiune 220V</li> <li>j) Programarea butoanelor aplicației Arduino Bluetooth Controller pentru a transmite în timp real comenzi către modulul de rele</li> </ul> <p>Programarea codului plăcii de prototipare pentru identificarea pinilor de comunicare cu senzorii și modulul Bluetooth</p>
------------------------	---

<b>Date experimentale și detalii esențiale ale experimentelor</b>	<p>Măsurătorile realizate în condiții atmosferice diferite (iarna și primăvara) au demonstrat că senzorii colectează datele în mod corect</p> <ul style="list-style-type: none"> <li>- Temperatură ambientală</li> <li>- Umiditatea aerului</li> <li>- Umiditatea solului</li> </ul> <p>După finalizarea prototipului se va finaliza produsul prin transferul componentelor la o placă de prototipare cu cablaj și sudură cu cositor.</p>
---	---

## Aplicația Arduino Bluetooth Controller și datele citite de senzori



## Concluzii

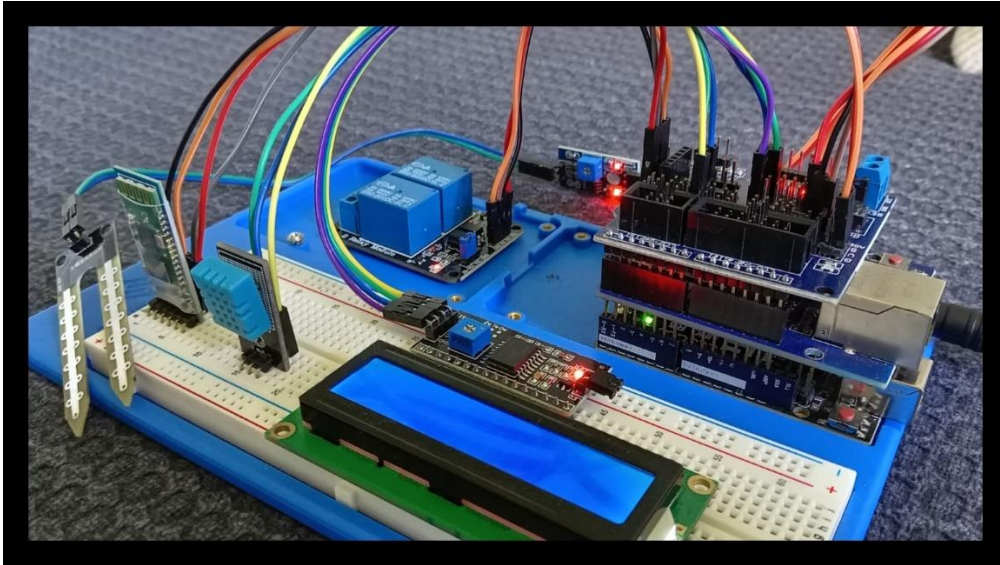
### Inovația și creativitatea proiectului iCASA constă în:

1. Utilizarea Scutului Ethernet pentru conectarea la Internet și stocarea datelor în cardul MicroSD
2. Utilizarea scutului pentru Senzori asigurând astfel fiabilitatea conexiunilor și optimizarea cablajului
3. Încorporarea aplicației mobile Arduino Bluetooth Controller pentru citirea în timp real a datelor colectate de senzori și transmiterea de mesaje pentru controlul Releelor
4. Combinarea plăcilor Arduino Uno , Ethernet și Sensor Shield și optimizarea codului

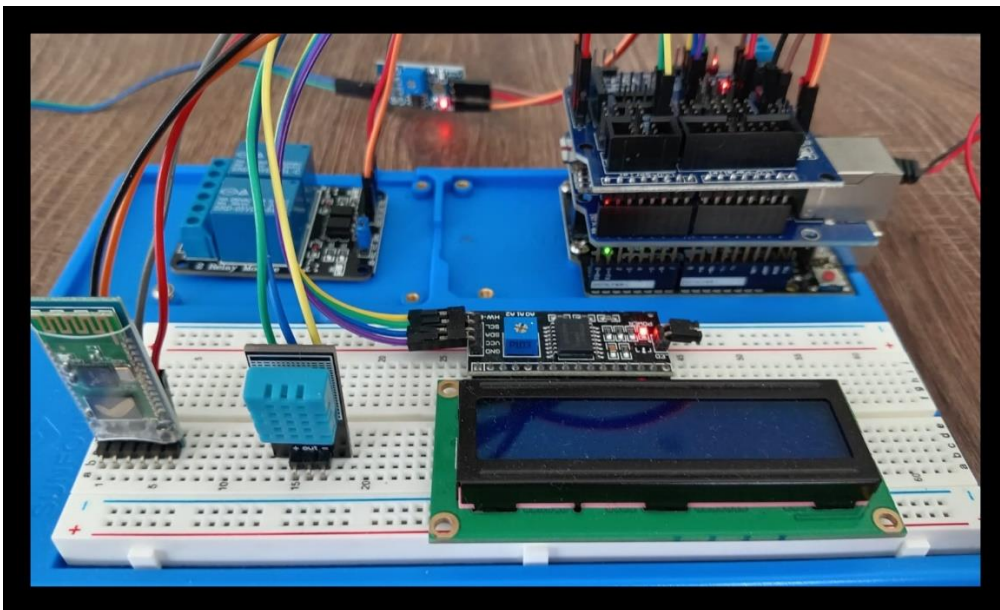
Proiectul este în curs de dezvoltare, funcționalitatea este testată și următoarele etape sunt planificate pentru îmbunătățire:

1. Finalizarea aplicației mobile OPERA și stabilizarea comunicării între aplicație și sistemul de control
2. Integrarea unui sistem de stocare a datelor istorice citite de senzori pentru analize ulterioare
3. Integrarea completă cu sistemul de irigație dezvoltat de echipa colaboratorilor externi

Anexe (imagini, scheme, cod)



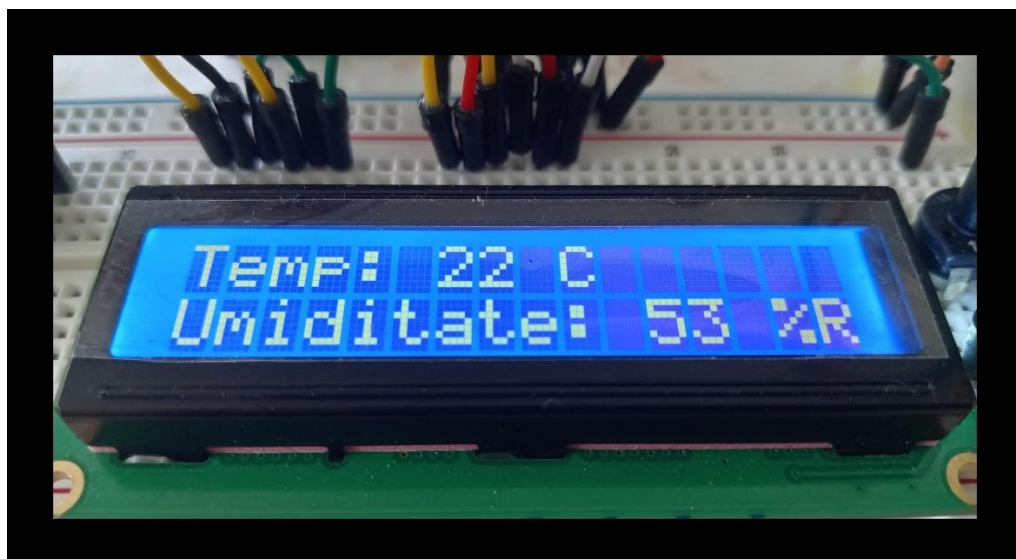
Imaginea 1



Imaginea 2

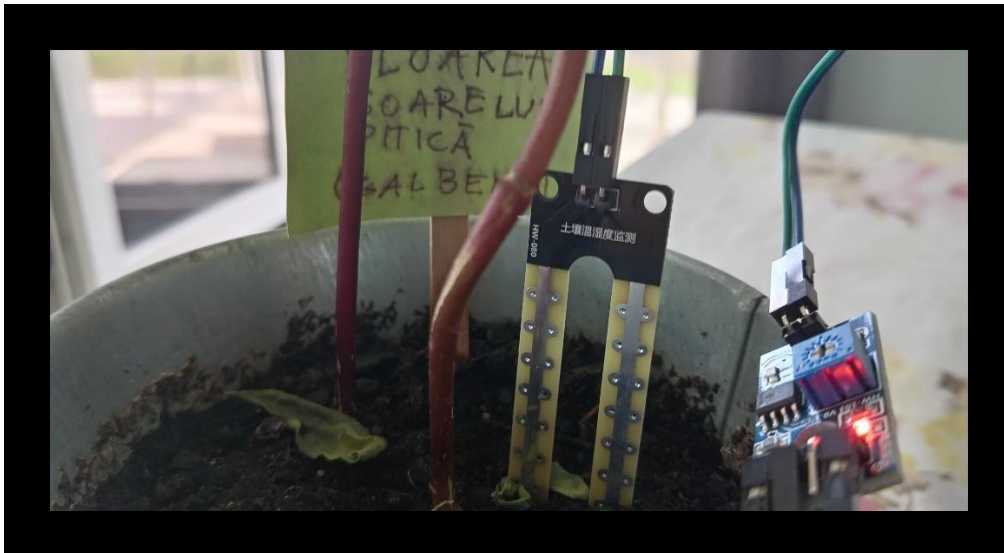


Imaginea 3

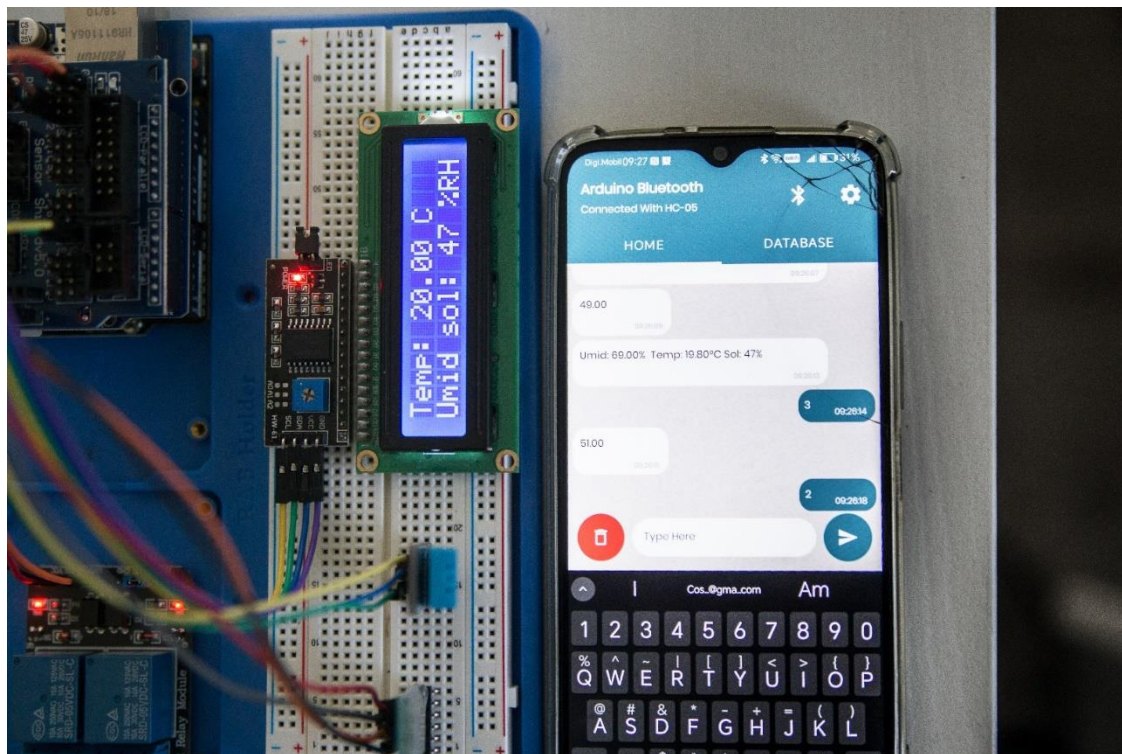


Imaginea 4

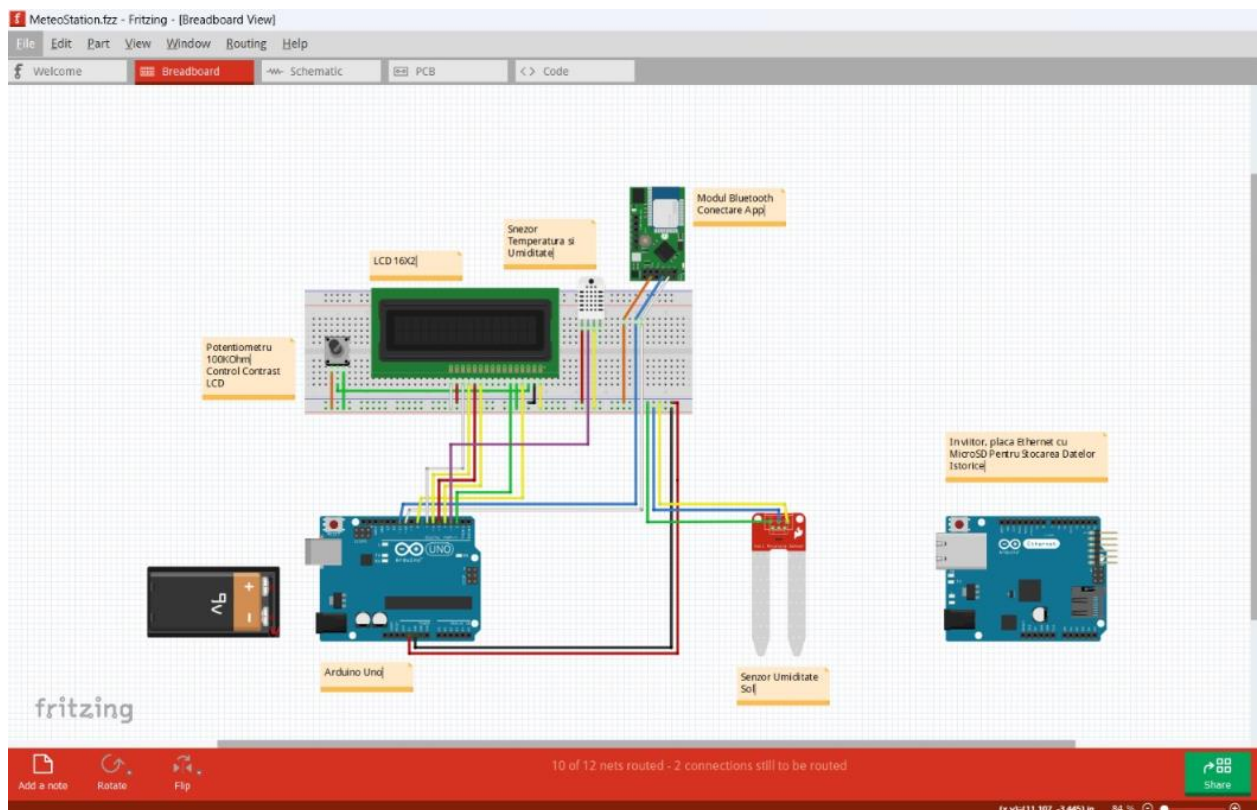




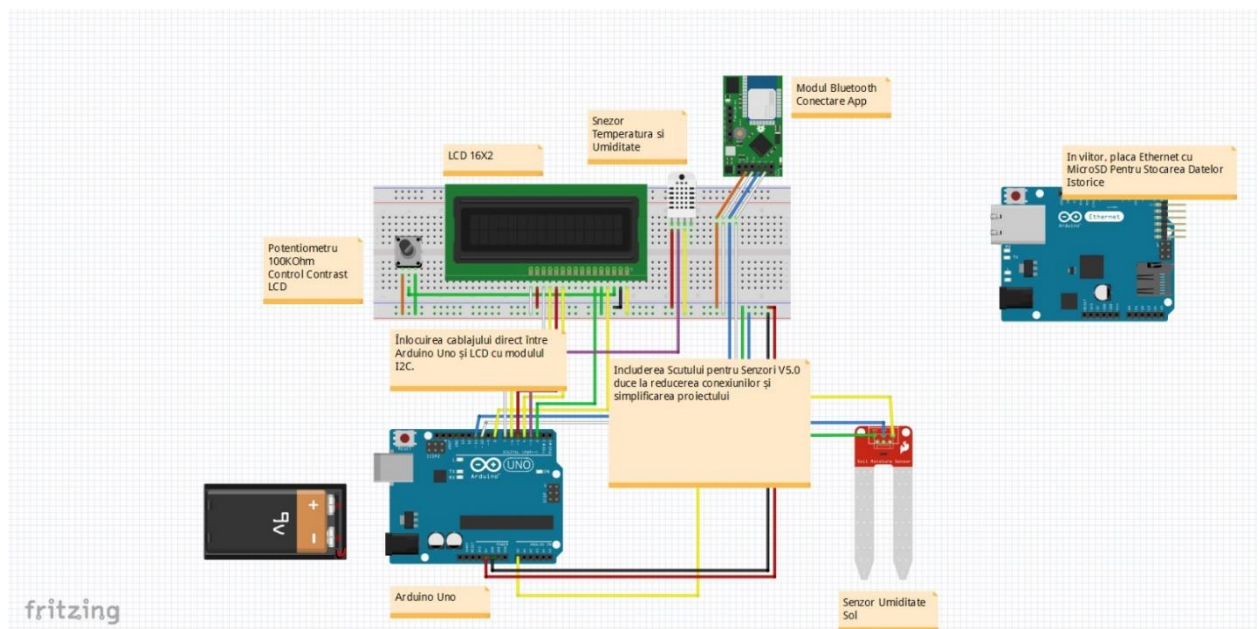
Imaginea 5



Imaginea 6



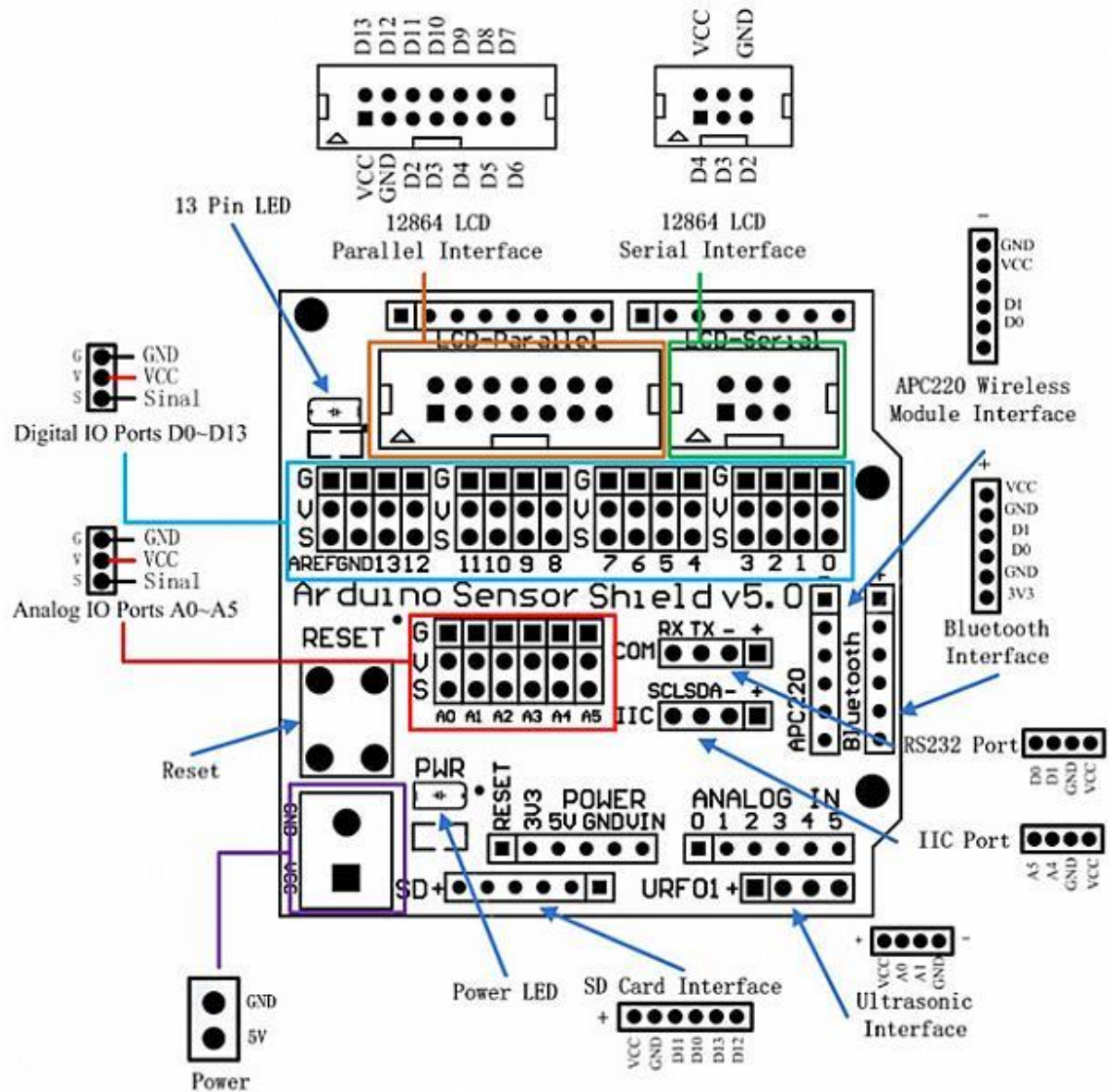
**Schema Prototip V1**



**Schema Prototip V2**



## Îmbunătățiri cablaj cu Scut pentru senzori



Arduino Sensor Shield v5.0 Functional Diagram

## Secvențe de cod din proiectul iCASA

```
Meteo_V8 | Arduino IDE 2.3.2
File Edit Sketch Tools Help
Arduino Uno

Meteo_V8.ino
19
20 #define RELAY 8
21
22 DHT dht(DHTPIN, DHTTYPE); //Initialize the DHT component
23
24 LiquidCrystal_I2C lcd(0x27, 16, 2);
25
26 int chk;
27 int umiditate;
28 int procent;
29
30 char t; // Variabila pentru mesaje catre releu
31
32 void setup() {
33
34     Serial.begin(9600);
35     Serial.println("DHTxx test!");
36
37     dht.begin();
38
39     // code setup, se va executa o singura data
40     lcd.init();
41     lcd.clear();
42     lcd.backlight(); // Porneste lumina de background a ecranului
43     lcd.begin(16,2);
44     lcd.print("METEO");
45     delay(1000);
46     lcd.clear();
47
48
49     pinMode(4, INPUT); //Pin Input Senzor DHT11
50     pinMode(A0, INPUT); //Pin Input Senzor Umiditate Sol
51 }
```

```
Meteo_V8 | Arduino IDE 2.3.2
File Edit Sketch Tools Help
Arduino Uno

Meteo_V8.ino
55
56 void afisare() { // Afisare Date Senzor DHT11
57
58     lcd.setCursor(0,0);
59     lcd.print("Temp: ");
60     lcd.print(dht.readTemperature());
61     lcd.print(" C");
62     lcd.setCursor(0,1);
63     lcd.print("Umiditate: ");
64     lcd.print(dht.readHumidity());
65     lcd.print(" %RH");
66     delay(2000);
67     lcd.clear();
68 }
69
70 void afisareSol() { // Afisare Date Senzor Umiditate Sol
71
72     lcd.setCursor(0,0);
73     lcd.print("Temp: ");
74     lcd.print(dht.readTemperature());
75     lcd.print(" C");
76     lcd.setCursor(0,1);
77     lcd.print("Umid sol: ");
78     lcd.print(procent);
79     lcd.print(" %RH");
80     delay(2000);
81     lcd.clear();
82 }
83 }
```

```
Meteo_V8 | Arduino IDE 2.3.2
File Edit Sketch Tools Help

Meteo_V8.ino
83
84 void loop() {
85
86 // Definitie text pentru transmitere catre APP
87 float h = dht.readHumidity();
88 float t = dht.readTemperature();
89
90
91 // Verifica daca lecturile senzorilor functioneaza. Daca nu, incearca din nou
92 if (isnan(h) || isnan(t)) {
93   Serial.println("Failed to read from DHT sensor!");
94   return;
95 }
96
97 umiditate = analogRead(0);
98 procent = 100 - (umiditate/1024.0) * 100;
99
100 afisare(); // Executa in loop afisare DHT11 pe LCD
101
102 afisareSol(); // Executa in loop afisare Senzor Sol pe LCD
103
104 // Comunica spre Android App prin Bluetooth
105
106 String message = (String) "Umid: "+h+"% Temp: " +t+"°C"+ " Sol: "+procent+"%";
107 Serial.println(message);
108 delay(2000);
109
110 // Cod pentru mesaje catre Releu de la Bluetooth
111
112 if(Serial.available()){
113   t = Serial.read();
114   Serial.println(t);
115 }
```