

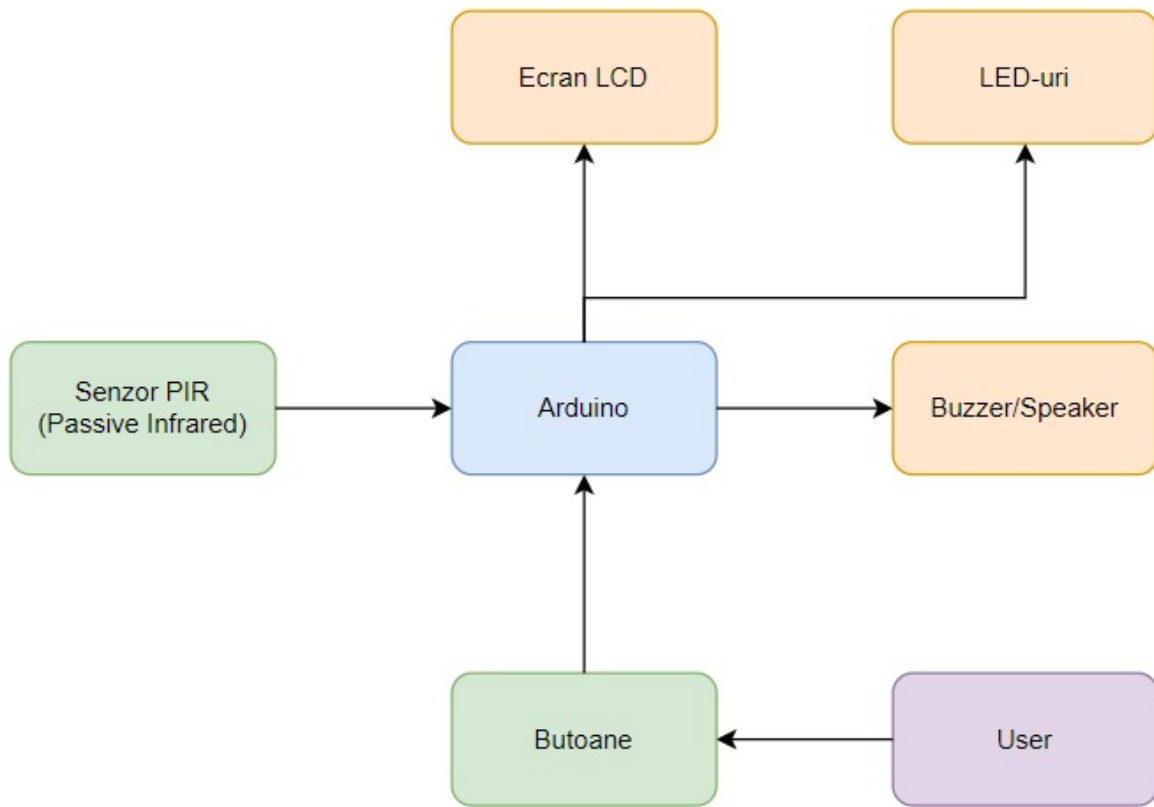
# Andreea-Gabriela Matei: Pomodoro Timer

## Introducere

- Proiectul presupune realizarea unui timer de tip Pomodoro. Ecranul se va aprinde atunci când se detectează mișcare. Utilizatorul poate alege tipul de timer cu ajutorul butoanelor. Atunci când timpul de lucru sau pauza expiră, buzzerul va semnaliza acest lucru. Un LED va sta aprins pentru timpul de lucru.
- Scop: eficientizarea lucrului și îmbunătățirea productivității
- Ideea de la care am pornit: nevoia de a gestiona timpul pe care îl petrec în fața ecranului și pauzele pe care le iau
- Proiectul este util pentru toți cei care vor să își gestioneze eficient timpul

## Descriere generală

- Ecranul se va aprinde atunci când este detectată mișcare
- Utilizatorul va interacționa cu programul folosind butoanele, cu ajutorul cărora va putea alege tipul de timer
- Pe ecran vor apărea diferite mesaje și timpul
- LED-urile vor fi folosite pentru semnalarea timpului de lucru
- Buzzerul semnalează expirarea timpului (pentru lucru sau pentru pauză)

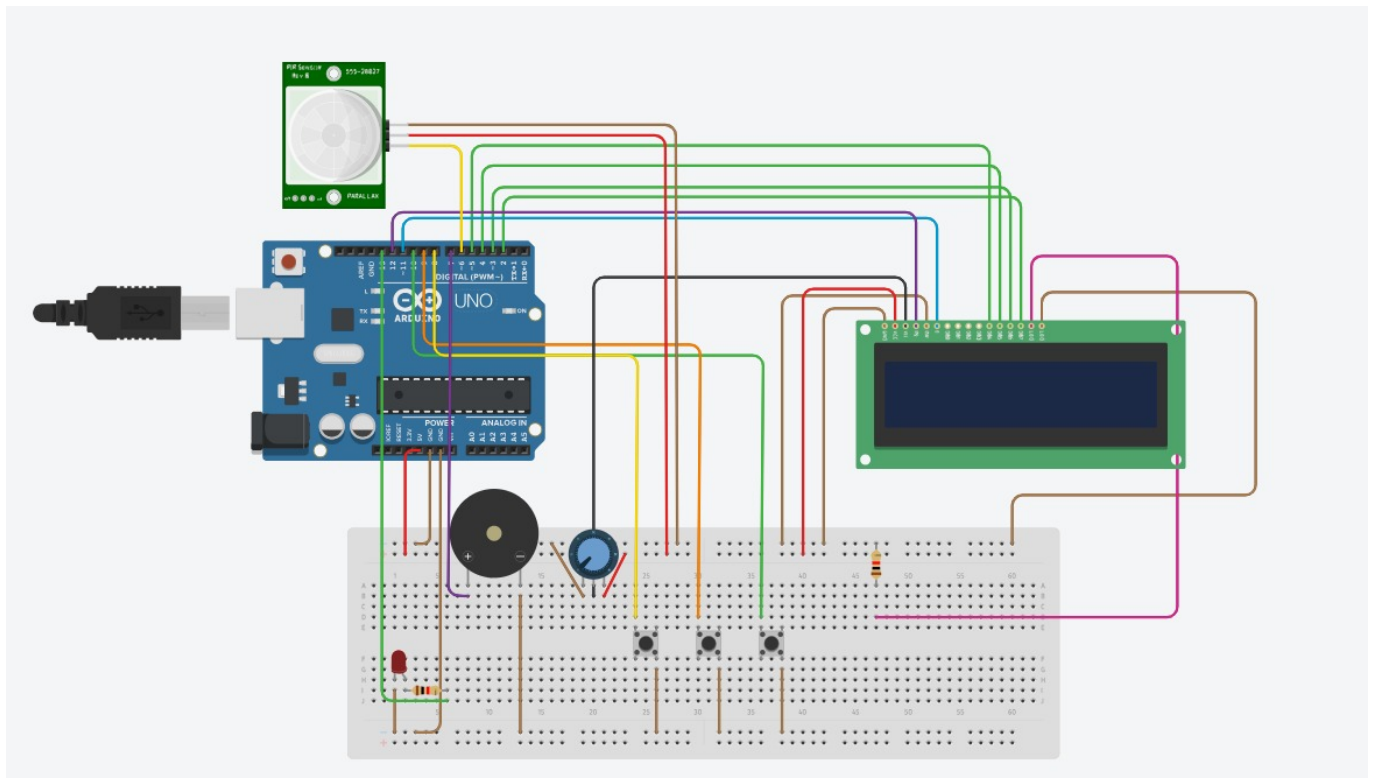


## Hardware Design

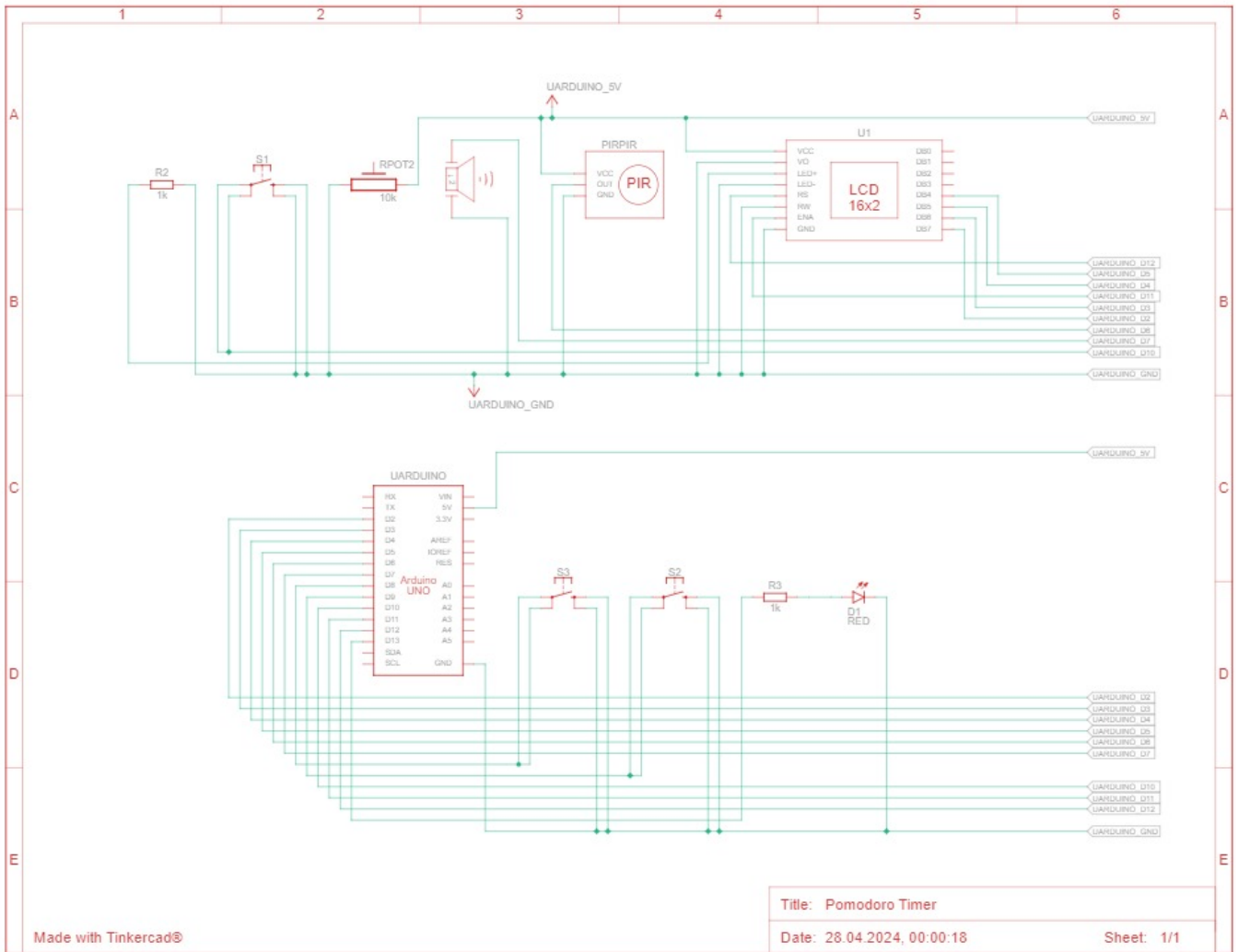
### Lista de piese

Name	Quantity	Component
UArduino	1	Arduino Uno R3
PIRPIR	1	PIR Sensor
U1	1	LCD 16 x 2
PIEZO1	1	Piezo
Rpot2	1	10 kΩ Potentiometer
S1 S2 S3	3	Pushbutton
R2 R3	2	1 kΩ Resistor
D1	1	Red LED

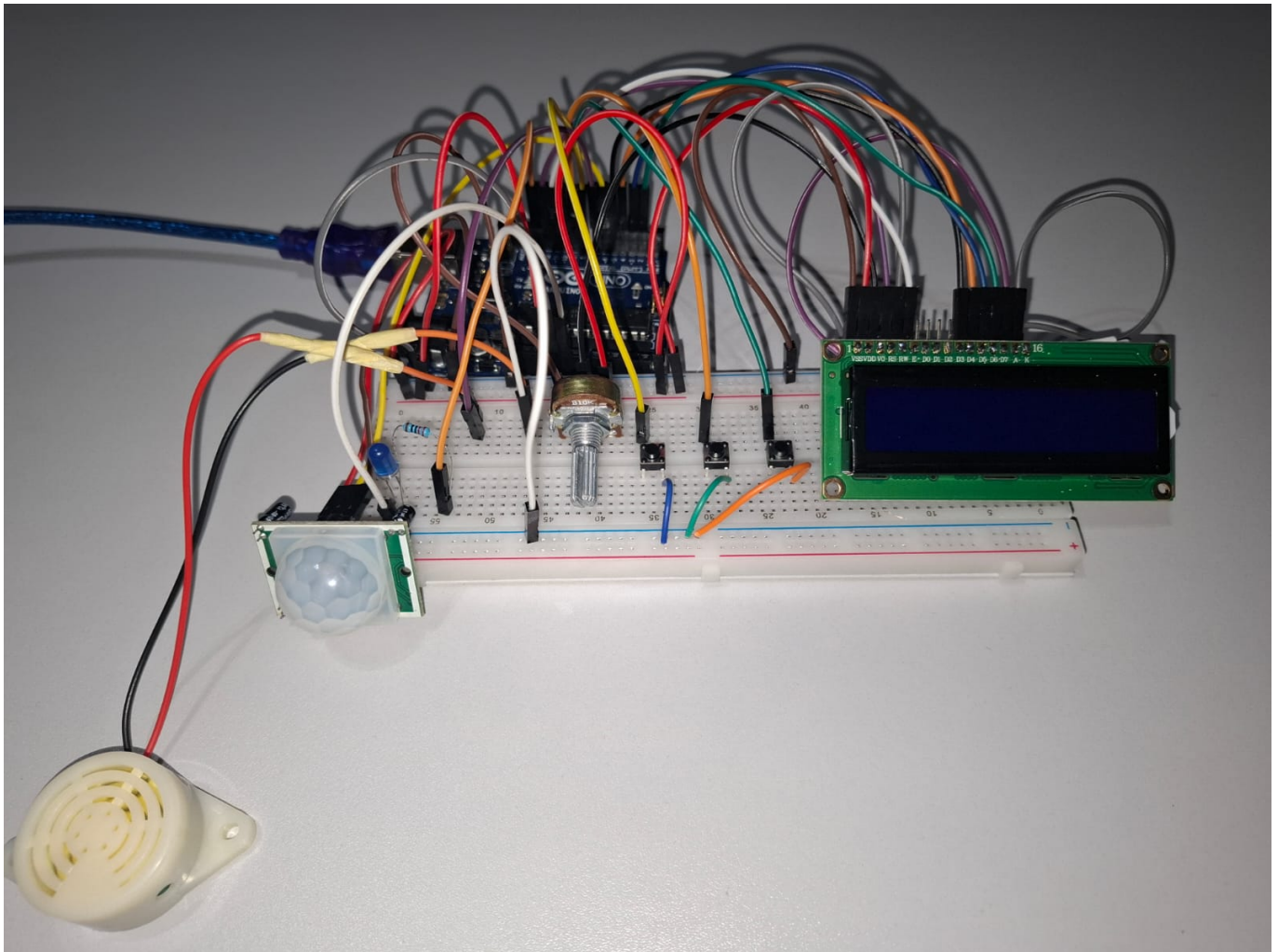
## Schema electrica



## Schematic View



## Implementare schema



## Software Design

### Descrierea codului aplicației

#### • Mediu de dezvoltare

- Arduino IDE: Codul este scris pentru microcontrolerul Arduino și utilizează biblioteca LiquidCrystal.h, specifică platformei Arduino.

#### • Librării și surse 3rd-party

- LiquidCrystal.h: Biblioteca standard Arduino pentru controlul ecranelor LCD. Aceasta permite interfațarea cu ecranele LCD folosind o configurație de pini specifică.

#### • Algoritmi și structuri implementate

- Controlul butoanelor: Codul implementează citirea butoanelor pentru navigarea prin meniuri și selectarea opțiunilor.
- Generarea tonurilor muzicale: Funcția mtone este utilizată pentru generarea de tonuri pe baza frecvenței specificate. Aceasta este folosită pentru a reda sunete la apăsarea butoanelor și pentru alerte.
- Gestionarea meniurilor: Codul conține mai multe funcții de actualizare a meniului (updateMenu, updateMenu2, updateMenu3) pentru a afișa diferite opțiuni pe LCD în funcție de starea curentă.
- Funcționarea pomodoro: Implementarea unui cronometru de 30 de minute pentru sesiuni de focus, cu alerte sonore la finalul fiecărei sesiuni și perioade de pauză.

- Detectarea mișcării: În cod, este un segment pentru detectarea mișcării folosind un senzor.

## Rezultate Obținute

Se poate viziona un video cu un demo al proiectului la acest link:

[https://youtu.be/LIF30h7-yOg?si=LIZzP8\\_S0mMvcp5B](https://youtu.be/LIF30h7-yOg?si=LIZzP8_S0mMvcp5B)

## Bibliografie/Resurse

- Pentru legarea ecranului la placa arduino: <https://docs.arduino.cc/learn/electronics/lcd-displays/>
- Pentru ideea proiectului: <https://projecthub.arduino.cc/edison0215/pomodoro-with-arduino-8bc0b4>
- Pentru generarea schemelor: <https://www.tinkercad.com/dashboard>
- Pentru folosirea senzorului PIR:  
<https://www.circuitbasics.com/how-to-setup-passive-infrared-pir-motion-sensors-on-the-arduino/>

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2024/ddosaru/andreea.matei2206>



Last update: **2024/05/27 12:55**