

ReMINDer

Introducere

“ReMINDer” este un dispozitiv ce vine in ajutorul oamenilor uituci ce vor sa duca la bun sfarsit toate task-urile, respectand deadline-urile acestora. Cu ajutorul lui “ReMINDer”, utilizatorul poate configura task-urile pe care le are, setand ziua si ora corespunzatoare acestora. De aici, “ReMINDer” va avea responsabilitatea de a semnala utilizatorul cu privinta la task-urile de indeplinit, in zilele si orele care au fost configurate.

Ideea de la care am pornit este aplicatia software “Reminders” de pe iOS, folosind-o in mod frecvent pentru task-urile de la facultate. Insa, uneori, fiind distras de multimea de notificari ale celorlalte aplicatii de pe telefon, nu observ notificarile privitoare la task-urile pe care le am de indeplinit. Asa ca m-am gandit ca ar fi mai util in a avea un dispozitiv separat pentru aceasta necesitate de a manageria timpul si task-urile avute.

Descriere generală

Schema bloc:



Utilizatorul va interactiona cu dispozitivul prin cele 4 butoane care sunt definite astfel: buton de back, un buton de next/confirm, 2 butoane de iterare (+/-) prin data, ora si alfabetul necesar setarii titlului

Cand data si ora sistemului vor face matching cu una din datele si orele unui task configurat de utilizator, sistemul va activa buzzerul, aprinderea ledului, pana in momentul in care utilizatorul apasa butonul “confirm”.

In afara momentelor in care sistemul se afla in starea de reminding (buzzerul e activat, ledul e aprins si display-ul afiseaza titlul task-ului), acesta afiseaza data, ora, temperatura si umiditatea (ultimele doua sunt preluate prin intermediul senzorului).

Hardware Design

Listă de piese:

- Arduino UNO
- 16x2 LCD Display
- 4 Button Board

- Buzzer
- Led
- senzor de temperatura si umiditate DHT11
- fire dupont
- motherboard

Software Design

Implementarea se foloseste de o configurare interna pe stari. Astfel, pentru fiecare stare, dispozitivul va avea un anumit comportament. Starea 0 este reprezentata de starea default atunci cand dispozitivul este in functiune, mai exact acesta va afisa pe LCD temperatura si ora. In functia loop() se vor asculta continuu eventualele intreruperi aparute prin apasarea butoanelor. Din starea 0, la apasarea unui buton, se ajunge in starea 1, si se initiaza configurarea unui nou reminder. In starea 2 se poate seta ora (2 variabile globale pentru ora si minut, care sunt incrementate, decrementate). De asemenea, in starea 0 se testeaza continuu daca una din orele reminderelor setate corespunde cu ora din prezent. Daca da, se intr-o stare in care buzzer-ul si ledul sunt activate, iar pe lcd este afisat titlul reminderului, pana cand, prin apasarea unui buton, dispozitivul se intoarce in starea. Toate remindererele setate vor fi salvate intr-un vector global.

Bibliografie/Resurse

Resurse Software

- Biblioteca folosita pentru redarea informatiilor pe LCD:
<https://www.arduino.cc/reference/en/libraries/liquidcrystal-i2c/>
- Biblioteca pentru folosirea ceasului de la laptop: <https://playground.arduino.cc/Code/Time/>
- Biblioteca pentru utilizarea senzorului de temperatura si umiditate DHT11:
<https://www.arduino.cc/reference/en/libraries/dht-sensor-library/>

Resurse Hardware

<https://cleste.ro/tastatura-nerica-4-1.html>

<https://cleste.ro/modul-buzzer-activ.html>

<https://cleste.ro/led-de-5-mm.html>

<https://cleste.ro/ecran-lcd-1602-iic-i2c.html>

<https://cleste.ro/senzor-digital-de-temperatura-i-umiditate-dht11-cu-led.html>

[Export to PDF](#)

From:

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

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2022/imacovei/reminder>



Last update: **2022/06/02 13:03**