|  | UNIVERSITATEA DIN CRAIOVA  FACULTATEA DE AUTOMATICĂ, CALCULATOARE ȘI ELECTRONICĂ  DEPARTAMENTUL DE AUTOMATICĂ ȘI ELECTRONICĂ |  |
| --- | --- | --- |

Monitorizarea cantității de CO2 dintr-un submarin

SAI 2.1

***Grupa:***

Gavrilescu Dragoș-Mihai

Vaduva Andrei-Mihai

Dragan Alin-Ionut

Bacanu Ana-Maria-Alexandra

Cuprins

[1. Cerințe de mediu](#_r7q541nf0k7f) 3

[1.1. Sistemul de operare](#_57t25mfr9cj) 3

[1.2 Standard](#_7kk569bnv9p3) 3

[2. Cerinte nonfunctionale](#_bmcesr1kpgmm) 3

[2.1. Securitate](#_oduu7ytrwljj) 3

[2.2. Uzabilitate](#_t7dsfkdh4z2k) 3

[3. Cazul de folosire](#_jdn7gy2dy86d) 4

[4. Cerințele Clientului](#_i046q2wrvs6k) 4

[5. Bibliografie](#_uxdsq2qs0vc) 5

# 1. Cerințe de mediu

## 1.1. Sistemul de operare

Arduino Integrated Development Environment (IDE) este o aplicație multiplatformă (pentru Windows, macOS, Linux) care este scrisă în funcții din C și C++. Este folosit pentru a scrie și încărca programe pe plăci compatibile Arduino, dar și, cu ajutorul nucleelor terțe, alte plăci de dezvoltare ale furnizorilor.

## 1.2 Standard

Standardul utilizat este Embedded C Coding Standard. Standardul de codificare C încorporat al Barr Group a fost dezvoltat pentru a minimiza erorile din firmware, concentrându-se pe reguli practice care țin deoparte erorile -- în același timp, îmbunătățind mentenabilitatea și portabilitatea software-ului încorporat.

# 2. Cerinte nonfunctionale

## 2.1. Securitate

Proiectul fiind desemnat implementării pe un submarin, acesta este realizat ca un circuit standalone nefiind necesar un strat de securitate.

## 2.2. Uzabilitate

Proiectul este proiectat sa ruleze în mod autonom, nu necesita intervenție umană.

# 3. Cazul de folosire

Acesta are un singur caz de utilizare, fiind proiectat doar sa asiste la filtrarea aerului de Co2 din interiorul unui submarin. Realizează aceasta functionalitate prin controlul unui ventilator care va filtra aerul în urma citirii nivelului de Co2 cu un traductor Co2. Vor fi avertizari specifice nivelului de Co2.

# 4. Cerințele Clientului

În acest capitol va fi prezentată o descriere detaliată a functionalitatii proiectului:

1. Valorile senzorilor sunt citite la intervale de o secundă.

2. Toate valorile citite trebuie sa fie într-un interval predefinit. Valorile în afara intervalului sunt ignorate.

3. Cand trei citiri succesive ale unui senzor generează valori în afara intervalului predefinit un led,ce corespunde senzorului respectiv, își schimba culoarea din verde în roșu (se genereaza ERROR).

4. În funcție de valorile citite se pot genera semnale WARNING sau ALARM.

5. Cand un semnal WARNING este generat un led își schimba culoarea din verde în roșu.

6. Un WARNING persista pana cand este achitat de pilot sau dispar condițiile care l-au generat.

7. După achitare un nou WARNING va fi generat doar dacă a existat cel puțin o citire pentru care nu s-a generat WARNING sau ALARM.

8. Cand un semnal ALARMĂ este generat un led își schimba culoarea din verde în roșu.

9. Un ALARM persista pana cand dispar condițiile care l-au generat.

10. Cantitatea de CO2 este citită cu ajutorul unui senzor.

11. Cantitatea de CO2 este afișată pe un ecran de tip ceas.

12. Atunci cand cantitatea de CO2 este este mai mare decat 0.1%, un semnal WARNING este generat.

13. Dacă semnalul WARNING este achitat atunci se porneste instalatia de ventilatie.

14. Atunci cand cantitatea de CO2 este este mai mare decat 0.3%, un semnal ALARMĂ este generat.

15. După apariția semnalului ALARM se porneste instalatia de ventilatie.

16. Instalatia de ventilatie se oprește dacă cantitatea de CO2 este mai mica decat 0.05%.

# 5. Bibliografie

* <https://en.wikipedia.org/wiki/Windows_10>
* <https://en.wikipedia.org/wiki/Arduino_IDE>
* https://barrgroup.com/embedded-systems/books/embedded-c-coding-standard#:~:text=Barr%20Group's%20Embedded%20C%20Coding,and%20portability%20of%20embedded%20software.&text=Individual%20rules%20that%20have%20been,types%20of%20bugs%20are%20highlighted.