2025-1-RO01-KA121-VET-000319082

4IT Erasmus+Project, 2025-1-RO01-KA121-VET-000319082

Beneficiary: "Gheorghe Vranceanu" National College, Bacau

Sending organization: "Roman Voda" National College, Roman

MOBILITY PORTFOLIO

Valencia, Spain/ Paphos/Cyprus

Mobility period 28.10.2025- 10.11.2025

Practice period: 29.10.2025-08.11.2025

RECEIVING COMPANY: LAZAROU ICT PROFESSIONALS

Practice tutor: Lazaros Lazarou

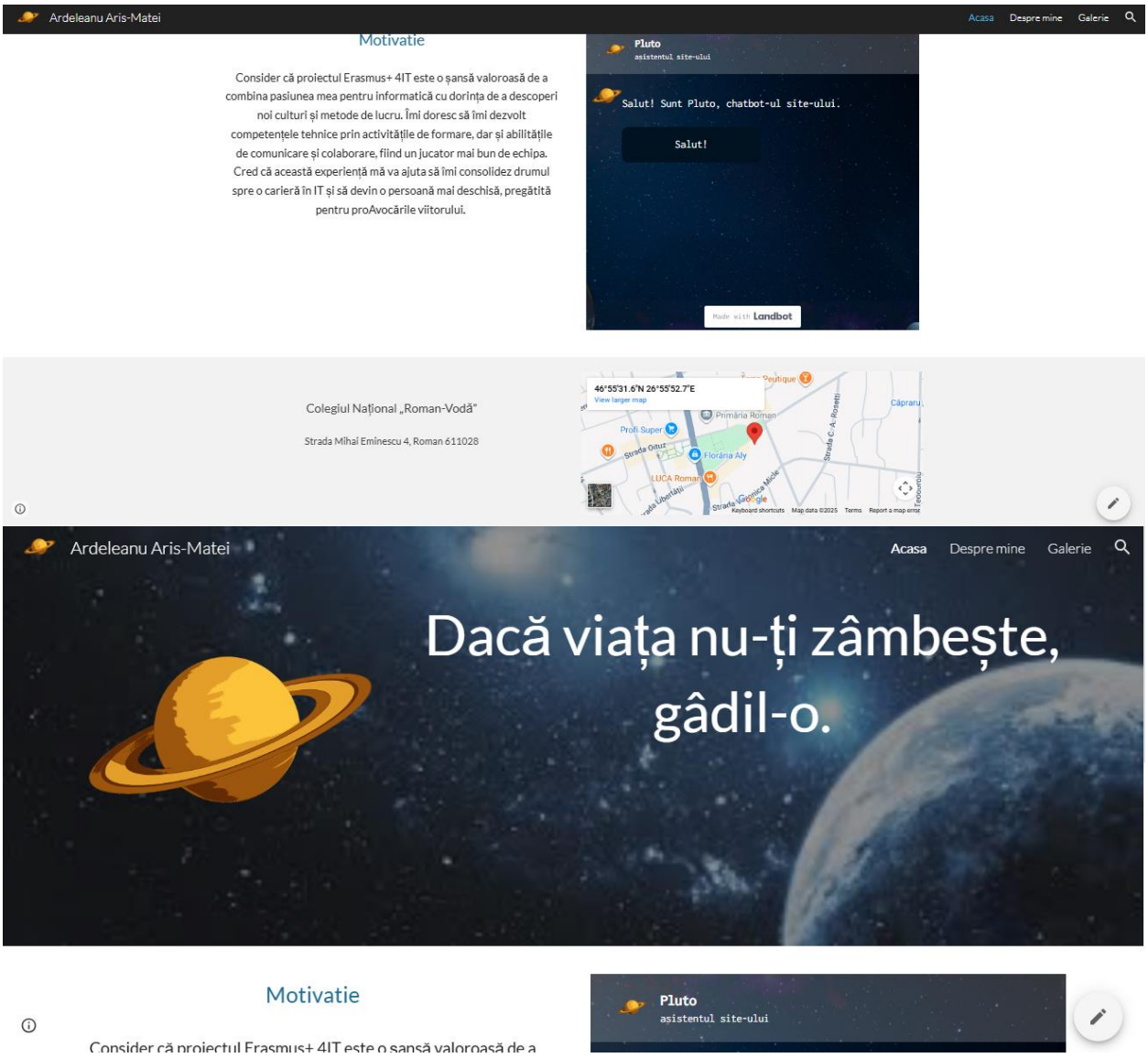
Participant: Ardeleanu Aris-Matei

DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

2025-1-RO01-KA121-VET-000319082

Ardeleanu Aris - selection site

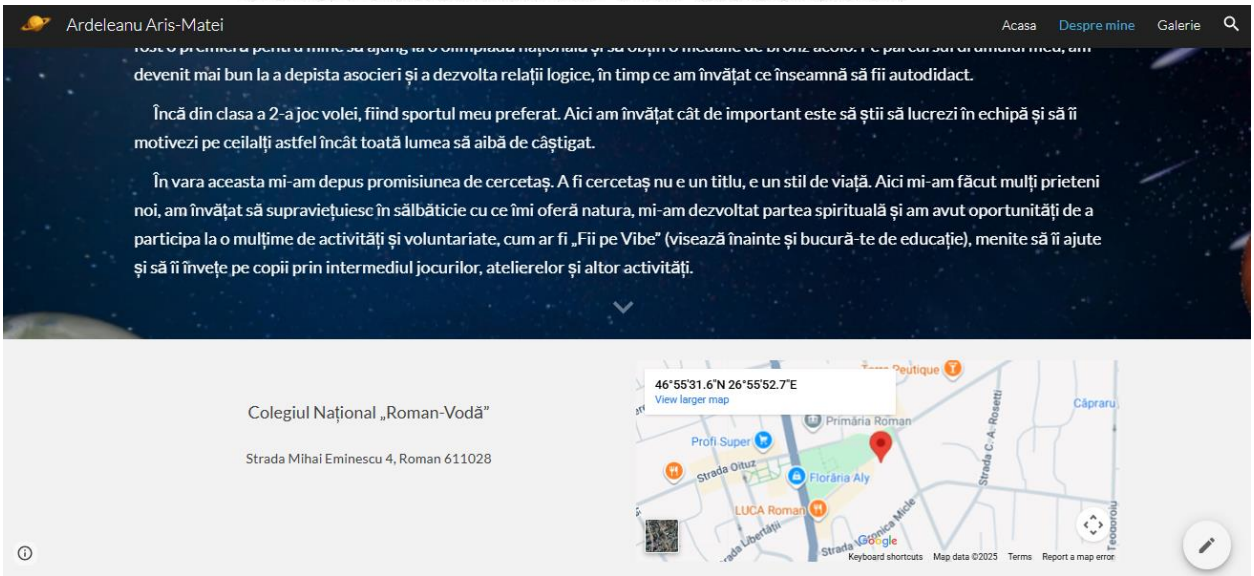


DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

Erasmus+

2025-1-RO01-KA121-VET-000319082

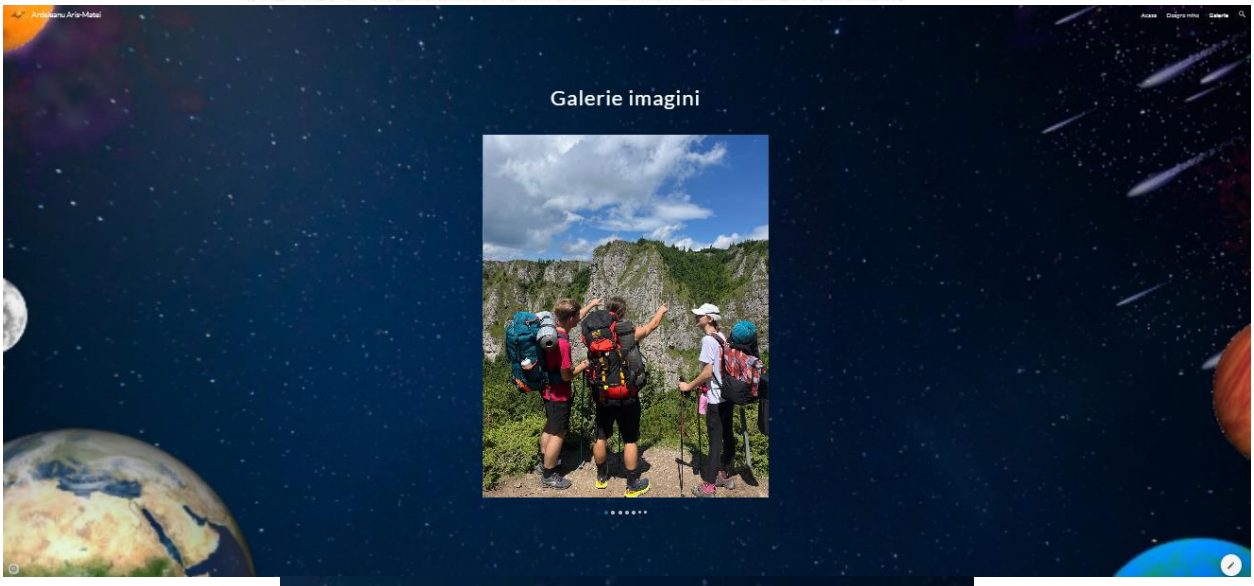


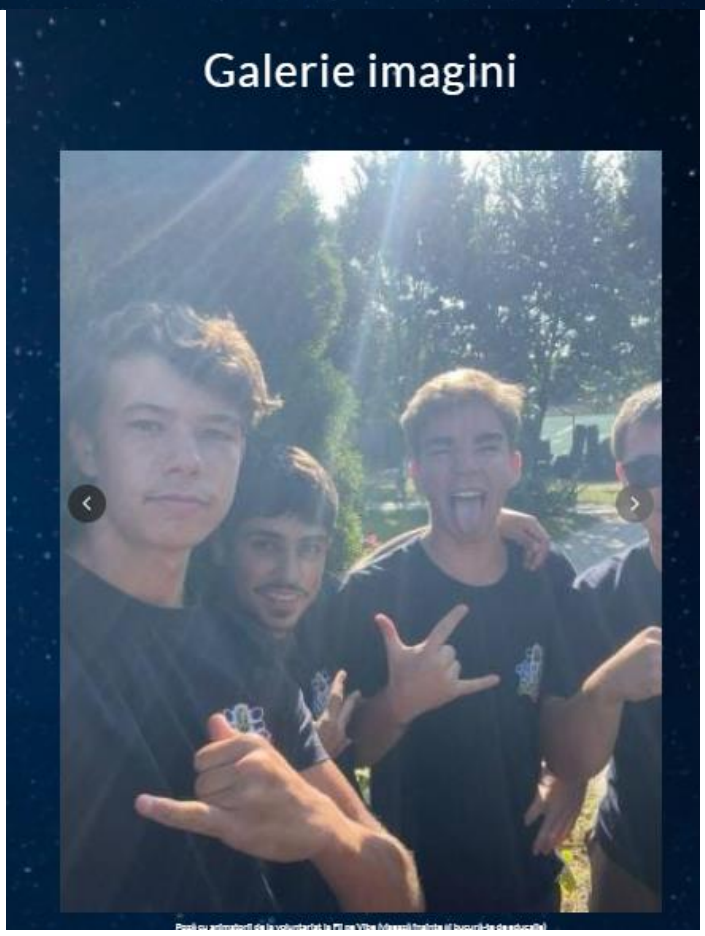


DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

2025-1-RO01-KA121-VET-000319082





DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

2025-1-RO01-KA121-VET-000319082

Cultural training project

CUPRINS

Pagina 1 – Titlu
Pagina 2 – Cuprins
Pagina 3 – Cipru în Evul Mediu
Pagina 4 – Perioada otomană și dominația britanică
Pagina 5 – Independența Republicii Cipru
Pagina 6 – Tradiții ale culturii cipriote
Pagina 7 – Personalități ale Ciprului
Pagina 8 – Concluzie
Pagina 9 – Bibliografie

2025-1-RO01-KA121-VET-000319082

CIPRU STAT INDEPENDENT

Ardeleanu Aris-Matei

DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

2025-1-RO01-KA121-VET-000319082

CIPRU ÎN EVUL MEDIU (SEC. XII-XVI)



https://checkedin.ro/de-la-pafos-la-nicosia-un-inceput-de-an-insorit/

- În Evul Mediu, Ciprul a fost o insulă importantă între Europa și Orient. Până în 1191 s-a aflat sub stăpânire bizantină, apoi a fost cucerit de regele Angliei Richard Inimă de Leu.
- În 1192, el a vândut insula cavalerului Guy de Lusignan, care a întemeiat Regatul Ciprului, condus de nobili francezi catolici. Populația locală era greacă și ortodoxă, dar nu avea drepturi politice.
- În 1489, Ciprul a fost anexat de Republica Venețiană, care a fortificat orașele principale pentru a se apăra de otomani. În 1571, insula a fost cucerită de Imperiul Otoman, sfârșind perioada medievală.

PERIOADA OTOMANĂ ȘI DOMINAȚIA BRITANICĂ (1571-1960)

După cucerirea otomană din 1571, Ciprul a devenit provincie a Imperiului Otoman. Biserica Ortodoxă a fost recunoscută oficial și a devenit principalul reprezentant al grecilor ciprioți.

În 1878, insula a fost dată în administrarea Marii Britanii, iar în 1914 a fost anexată oficial. Din 1925 a devenit colonie britanică.

Majoritatea grecilor ciprioți doreau unirea cu Grecia (enosis), iar în 1955 organizația EOKA a început o luptă armată împotriva britanicilor. Turcii ciprioți, sprijiniți de Turcia, s-au opus unirii și au cerut împărțirea insulei (taksim).



https://history-maps.com/ro/story/History-of-the-Ottoman-Empire

DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

2025-1-RO01-KA121-VET-000319082

INDEPENDENȚA REPUBLICII CIPRU (1960)



După ani de tensiuni și negocieri între Grecia, Turcia și Marea Britanie, s-au semnat Acordurile de la Zürich și Londra (1959), care au stabilit independența Ciprului.

La 16 august 1960, a fost proclamată Republica Cipru, cu președinte Makarios al III-lea și vicepreședinte Fazıl Küçük.

Noua constituție împărțea puterea între grecii și turcii ciprioți. Independența a încheiat peste șapte secole de dominație străină și a marcat începutul unei noi etape pentru poporul cipriot.

TRADIȚII ALE CULTURII CIPRIOTE

Cultura Ciprului combină influențe grecești, turcești și orientale, reflectând istoria diversă a insulei. Muzica tradițională folosește instrumente precum lăuta și bouzouki, iar dansurile populare, precum sgrtos și zeibekiko, se joacă la sărbători.

Portul tradițional este colorat, cu broderii și țesături fine lucrate manual. Bucătăria cipriotă este renumită pentru mâncăruri precum halloumi (brânză la grătar), souvlaki (friptură) și meze (gustări variate).

Sărbătorile religioase, în special Paștele ortodox, sunt cele mai importante și se marchează prin procesiuni, slujbe de noapte și mese în familie. O altă tradiție populară este ospitalitatea – cipriotii sunt cunoscuți pentru modul cald și prietenos de a-și primi oaspeții.

DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

2025-1-RO01-KA121-VET-000319082

Personalități importante ale Ciprului

Arhiepiscopul Makarios al III-lea (1913–1977)

A fost primul președinte al Republicii Cipru după obținerea independenței în 1960. Makarios a avut un rol esențial în mișcarea pentru eliberarea de sub dominația britanică și a susținut conviețuirea pașnică între grecii și turcii ciprioți. A fost o figură respectată atât ca lider religios, cât și ca om politic.





George Seferis (1900–1971)

Poet de origine cipriotă, laureat al Premiului Nobel pentru Literatură în 1963. Opera sa exprimă dragostea pentru limba greacă, tradiția elenă și valorile umane universale. A contribuit la promovarea culturii cipriote și grecești pe plan internațional.

https://ro.wikipedia.org/wiki/Giorgos_Seferis

Concluzie



https://www.slidegeeks.com/cyprus-country-powerpoint-flags

Cipru are o istorie lungă, marcată de multe stăpâniri străine, dar poporul său și-a păstrat identitatea și cultura.

Independența din 1960 a însemnat începutul unei noi etape pentru insulă.

Tradițiile, muzica și personalitățile sale arată legătura dintre Orient și Occident și dorința cipriotilor de libertate și unitate.

DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

2025-1-RO01-KA121-VET-000319082



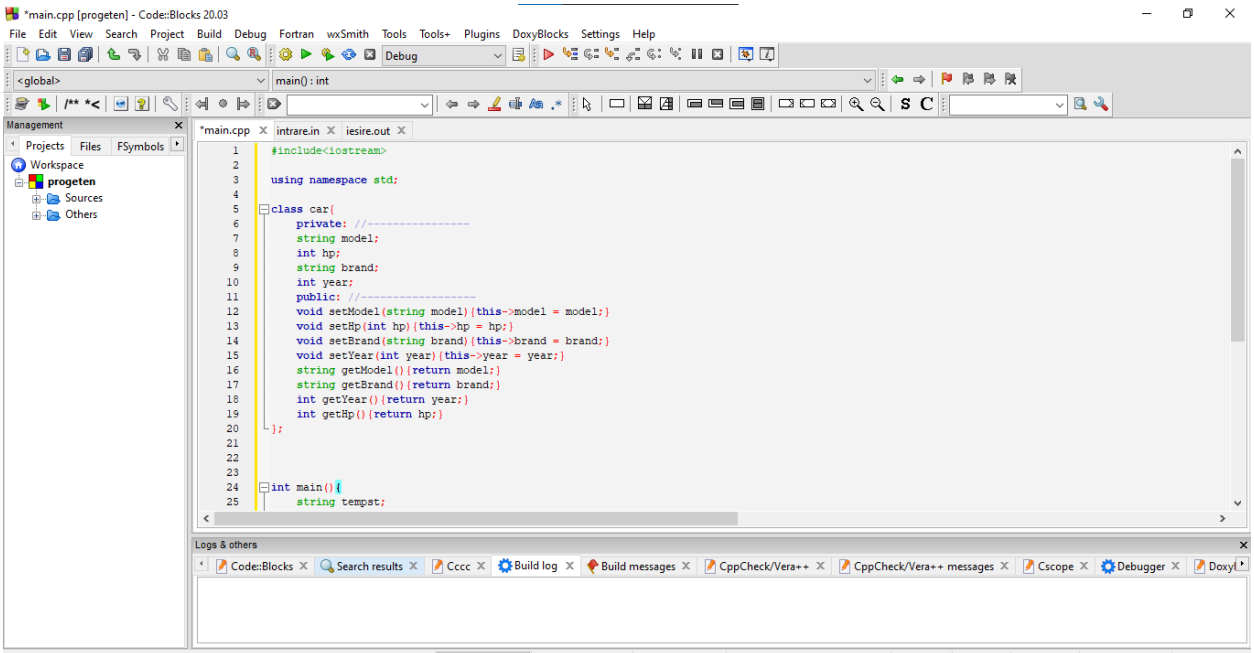
DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

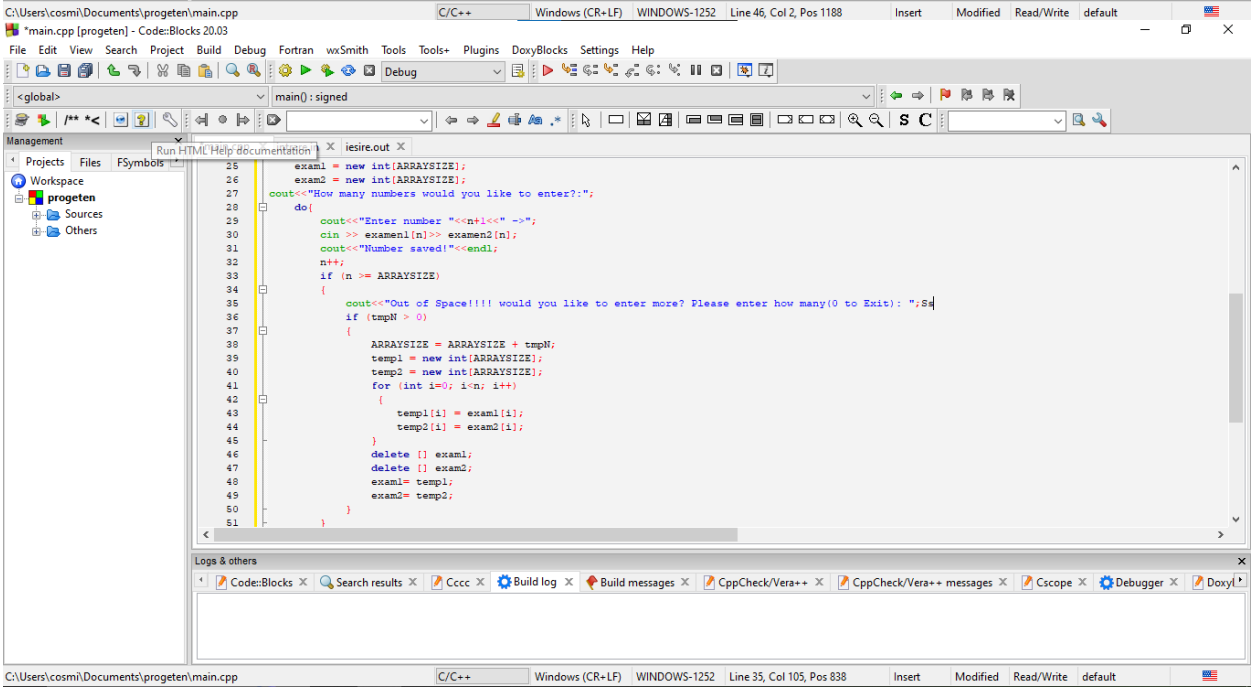
Co-funded by the Erasmus+ Programme of the European Union

Erasmus+

2025-1-RO01-KA121-VET-000319082

C++ Coding Projects



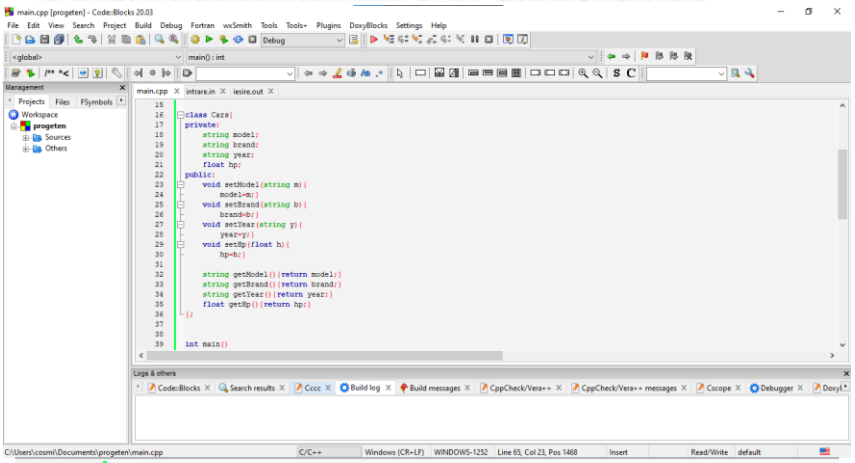


DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

Erasmus+

2025-1-RO01-KA121-VET-000319082



```cpp
#include <iostream>
using namespace std;

class patient {
private:
    string name;
    int age;
    string phone;
    string medicine;

public:
    void setName(string n) { name = n; }
    void setAge(int a) { age = a; }
    void setPhone(string p) { phone = p; }
    void setMedicine(string m) { medicine = m; }

    string getName() { return name; }
    int getAge() { return age; }
    string getPhone() { return phone; }
    string getMedicine() { return medicine; }
};

// Functii auxiliare
void addPatient(patient *cl, int index);
patient *growArray(patient *cl, int arraySize, int index);
void printArray(patient *clinic, int index);
void searchByName(patient *clinic, int index, string name);
void searchByMedicine(patient *clinic, int index, string medicine);

int main() {
    int ArraySize = 1;
    int index = 0;
    patient *clinic;
    clinic = new patient[ArraySize];
    int userChoice = 1;

    while (userChoice > 0) {
        cout << "1. Add a new user" << endl;
        cout << "2. Edit a User" << endl;
        cout << "3. Delete a User" << endl;
        cout << "4. Search by Name" << endl;
```

DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

Erasmus+

2025-1-RO01-KA121-VET-000319082

```
        else if (userChoice == 0) {
            cout << "Thank you for using the program!" << endl;
        }
        else {
            cout << "Wrong Choice! Try again" << endl;
        }
    }

    delete[] clinic;
    return 0;
}

patient *growArray(patient *clinic, int ArraySize, int index) {
    patient *tmp;
    tmp = new patient[ArraySize];
    for (int i = 0; i < index; i++) {
        tmp[i].setAge(clinic[i].getAge());
        tmp[i].setMedicine(clinic[i].getMedicine());
        tmp[i].setName(clinic[i].getName());
        tmp[i].setPhone(clinic[i].getPhone());
    }
    delete[] clinic;
    return tmp;
}

void addPatient(patient *cl, int index) {
    string name, phone, med;
    int age;

    cout << "Name: ";
    cin >> name;
    cl[index].setName(name);
    cout << "Age: ";
    cin >> age;
    cl[index].setAge(age);
    cout << "Phone: ";
    cin >> phone;
    cl[index].setPhone(phone);
    cout << "Medicine Given: ";
    cin >> med;
    cl[index].setMedicine(med);
```

```
        cout << "4. Search by Name" << endl;
        cout << "5. Search by Medicine" << endl;
        cout << "6. Print The Array" << endl;
        cout << "0. Exit" << endl;
        cout << "What would you like to do? ";
        cin >> userChoice;
        cout << "--------------------------------" << endl;

        if (userChoice == 1) {
            cout << "Add a new Patient" << endl << "-----------------" << endl;
            if (index >= ArraySize) {
                ArraySize = ArraySize + 1;
                clinic = growArray(clinic, ArraySize, index);
            }
            addPatient(clinic, index);
            index++;
        }
        else if (userChoice == 2) {
            cout << "Edit a Patient (not implemented yet)" << endl;
        }
        else if (userChoice == 3) {
            cout << "Delete a Patient (not implemented yet)" << endl;
        }
        else if (userChoice == 4) {
            cout << "Search By Name" << endl << "-----------------" << endl;
            string name;
            cout << "Enter name to search: ";
            cin >> name;
            searchByName(clinic, index, name);
        }
        else if (userChoice == 5) {
            cout << "Search By Medicine" << endl << "-----------------" << endl;
            string med;
            cout << "Enter medicine to search: ";
            cin >> med;
            searchByMedicine(clinic, index, med);
        }
        else if (userChoice == 6) {
            printArray(clinic, index);
        }
        else if (userChoice == 0) {
```

DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

```
111         cin >> name;
112         cl[index].setName(name);
113         cout << "Age: ";
114         cin >> age;
115         cl[index].setAge(age);
116         cout << "Phone: ";
117         cin >> phone;
118         cl[index].setPhone(phone);
119         cout << "Medicine Given: ";
120         cin >> med;
121         cl[index].setMedicine(med);
122         cout << "Patient Successfully Added!" << endl;
123     }
124
125     void printArray(patient *clinic, int index) {
126         cout << "PRINTING ALL DATA" << endl << "----------------" << endl;
127         for (int i = 0; i < index; i++) {
128             cout << "----- Patient " << i + 1 << " -----" << endl;
129             cout << "Name: " << clinic[i].getName() << endl;
130             cout << "Age: " << clinic[i].getAge() << endl;
131             cout << "Phone: " << clinic[i].getPhone() << endl;
132             cout << "Medicine: " << clinic[i].getMedicine() << endl;
133         }
134     }
135
136     /*-------------------------------------
137         SEARCH FUNCTIONS
138     -------------------------------------*/
139
140     void searchByName(patient *clinic, int index, string name) {
141         bool found = false;
142         for (int i = 0; i < index; i++) {
143             if (clinic[i].getName() == name) {
144                 cout << "Patient Found!" << endl;
145                 cout << "Name: " << clinic[i].getName() << endl;
146                 cout << "Age: " << clinic[i].getAge() << endl;
147                 cout << "Phone: " << clinic[i].getPhone() << endl;
148                 cout << "Medicine: " << clinic[i].getMedicine() << endl;
149                 found = true;
150             }
151         }
152         if (!found)
153             cout << "No patient found with the name \"" << name << "\"." << endl;
154     }
155
156     void searchByMedicine(patient *clinic, int index, string medicine) {
157         bool found = false;
158         for (int i = 0; i < index; i++) {
159             if (clinic[i].getMedicine() == medicine) {
160                 cout << "Patient Found!" << endl;
161                 cout << "Name: " << clinic[i].getName() << endl;
162                 cout << "Age: " << clinic[i].getAge() << endl;
163                 cout << "Phone: " << clinic[i].getPhone() << endl;
164                 cout << "Medicine: " << clinic[i].getMedicine() << endl;
165                 found = true;
166             }
167         }
168         if (!found)
169             cout << "No patient found taking \"" << medicine << "\"." << endl;
170     }
171
```

DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

2025-1-RO01-KA121-VET-000319082

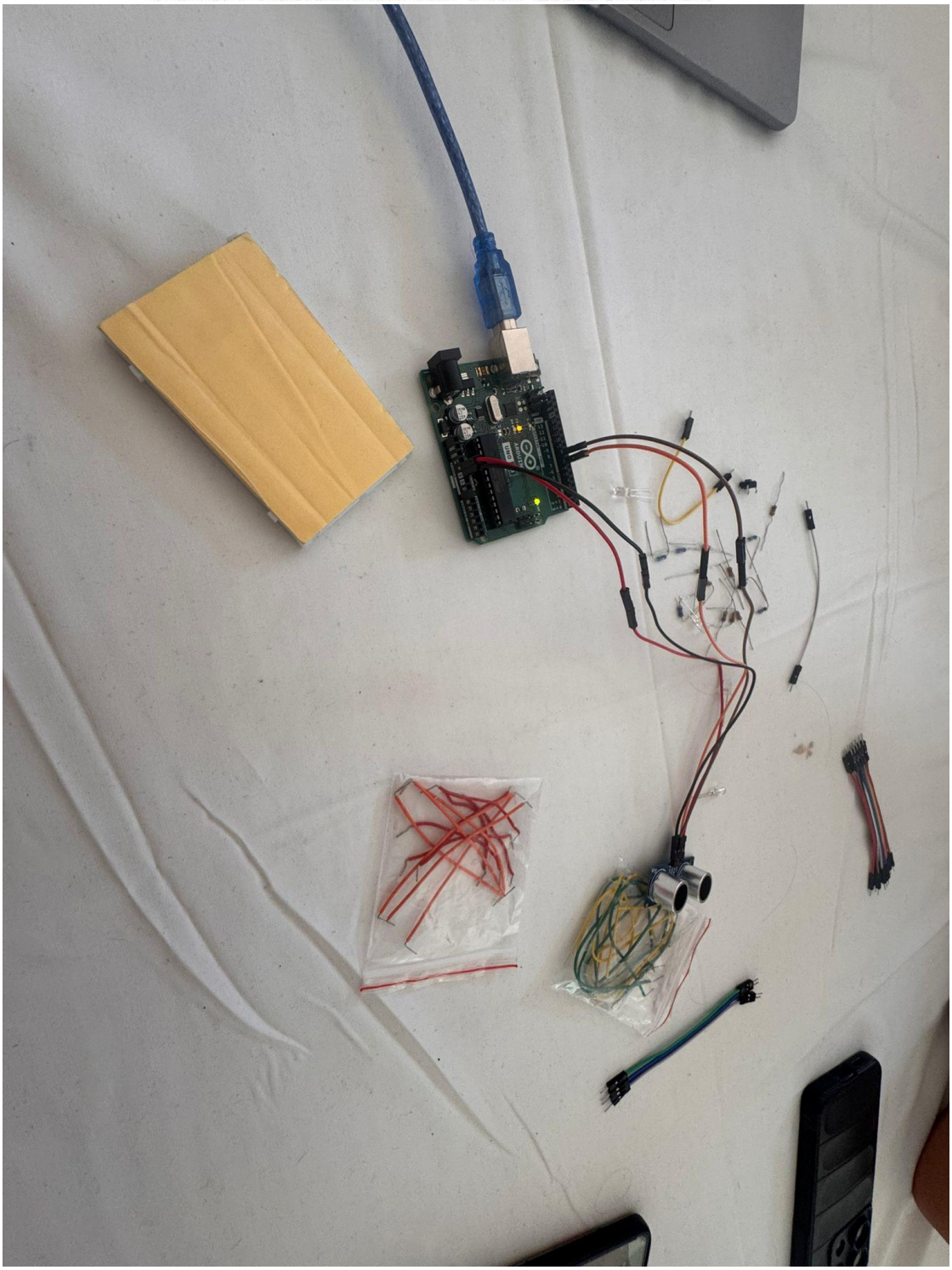
Arduino



DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Co-funded by the Erasmus+ Programme of the European Union

2025-1-RO01-KA121-VET-000319082



DISCLAIMER: The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.