

**Liceul “Ștefan Procopiu” Vaslui**

**LUCRARE PENTRU OBȚINEREA  
ATESTATULUI PROFESIONAL  
= profil matematică – informatică =**

**CALENDAR EVENIMENTE CU ARDUINO**

**Profesor îndrumător: Bărdiță Carmen**

**Absolvent: Mandrea Radu-Alexandru**

**- 2022 -**

## Introducere

Proiectul *Calendar evenimente cu Arduino* are caracter interdisciplinar, îmbinând competențe din domeniul informaticii, în special, cu fizica și proiectarea 3D, pentru a asigura caracterul practic al lucrării. Lucrarea conține 3 capitole structurate astfel:

Capitolul I – *Arduino* conține 5 subcapitole (*Despre Arduino, Istoric Arduino, Hardware, Plăcuțe oficiale Arduino + shield-uri, Software*). Acest capitol reflectă partea teoretică a lucrării, rezultată din documentarea pe site-uri de specialitate, având ca scop sublinierea importanței limbajului de programare Arduino și utilitatea acestuia în diverse domenii, Arduino fiind un limbaj de maximă importanță și eficacitate în domeniile medicină, mecanică, electronică, robotică.

Următoarele două capitole tratează partea practică a lucrării de atestat și anume site-ul web, respectiv produsul final auxiliar, Calendar Evenimente Arduino.

Capitolul al II – lea, *Site web* conține două subcapitole (*Site Web, Arhitectura Site-ului Web*). Primul subcapitol prezintă limbajele de programare folosite în crearea site-ului (HTML), dar și în stilizarea acestuia, modificând schema de culori (CSS), dar și adăugând unele animații (Javascript). Al doilea subcapitol descrie fiecare pagină din site cu accent pe etapele consecutive care corespund actualizării site-ului cu informații. Ca rezultat final, site-ul nu are doar rolul de a prezenta funcționalitatea calendarului cu Arduino, ci poate fi folosit drept un suport privind modul de utilizare a proiectului. Site-ul este atractiv, bine structurat, și facilitează utilizarea intuitivă.

Capitolul al III – lea este dedicat în întregime produsului final auxiliar al proiectului și anume *Calendarul de evenimente cu Arduino*, care a fost realizat cu ajutorul componentelor electronice și al imprimantei 3D. Capitolul se axează pe funcționarea proiectului, descriind etapele programării pas cu pas.

În concluzie, considerăm că lucrarea de atestat „Calendar Evenimente cu Arduino” este un proiect complex care atrage atenția asupra legăturii între domenii și necesitatea expunerii acestei legături în cadrul orelor de informatică. Un avantaj al lucrării este faptul că poate fi folosită ca bază pentru o dezvoltare ulterioară, în învățământul superior sau în cercurile de robotică care sunt foarte căutate de către elevii de gimnaziu și liceu.

# Capitolul I – Arduino

## I.1 Despre Arduino

**Arduino** este o companie open-source care produce atât plăcuțe de dezvoltare bazate pe microcontrolere, cât și partea de software destinată funcționării și programării acestora. Pe lângă acestea include și o comunitate uriașă care se ocupă cu creația și distribuirea de proiecte care au ca scop crearea de dispozitive care pot sesiza și controla diverse activități sau procese în lumea reală.

Proiectul este bazat pe designul plăcilor cu microcontroler produse de câțiva furnizori, folosind diverse tipuri de microcontrolere. Aceste plăci pun la dispoziția utilizatorului pini I/O, digitali și analogici, care pot fi interfațați cu o gamă largă de plăcuțe numite scuturi (shield-uri) și/sau cu alte circuite. Plăcile au interfețe de comunicații seriale, inclusiv USB pe unele modele, pentru a încărca programe din calculatoarele personale. Pentru programarea microcontrolerelor, Arduino vine cu un mediu de dezvoltare integrat (IDE) bazat pe proiectul Processing, care include suport pentru limbaje de programare ca C și C++.

Primul Arduino a fost lansat în 2005, având ca țintă asigurarea unei soluții ieftine și simple pentru începători și profesioniști spre a crea dispozitive capabile să interacționeze cu mediul, folosind senzori și sisteme de acționare. Cele mai comune exemple sunt dispozitivele pentru utilizatorii începători precum: roboții simpli, termostatele și/sau detectoarele de mișcare.

Plăcuțele Arduino sunt disponibile comercial sub formă preasamblată sau sub forma unor kituri de asamblat acasă (do-it-yourself). Specificațiile schemelor sunt disponibile pentru orice utilizator, permițând oricui să fabrice plăcuțe Arduino. Adafruit Industries estimase la mijlocul anului 2011 că peste 300.000 de plăcuțe oficiale Arduino au fost produse, iar în 2013 700.000 de plăcuțe oficiale erau în posesia utilizatorilor.

## I.2 Istoric Arduino

Arduino a început în 2005 ca un proiect al unui student al Institutului de Interacțiune a Designului din Ivrea, Italia. La acea vreme studenții foloseau o plăcuță de dezvoltare BASIC

Stamp care costa 100 de dolari, ceea ce era considerat foarte scump pentru studenți. Massimo Banzi, unul dintre fondatori, era student la Ivrea. Numele "Arduino" provine de la un bar din Ivrea, locul unde o parte din fondatori obișnuiau să se întâlnească.

Studentul columbian Hernando Barragán a creat platforma de dezvoltare Wiring care a servit ca bază pentru Arduino. După finalizarea platformei Wiring, mai multe versiuni, mai light și mai ieftine, au fost create și puse la dispoziția comunităților open-source. Din echipa inițială Arduino au făcut parte Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino și David Mellis.

## **I.3 Hardware**

O plăcuță Arduino este compusă dintr-un microcontroler Atmel AVR de 8-, 16- sau 32-biți (deși începând cu 2015 s-au folosit microcontrolere de la alți producători) cu componente complementare care facilitează programarea și încorporarea în alte circuite. Un aspect important la Arduino este că acesta dispune de conectori standard, care permit utilizatorului să conecteze plăcuța cu procesorul la diferite module interschimbabile numite shield-uri. Unele shield-uri comunică cu Arduino direct prin pinii digitali sau analogici, dar altele sunt adresabile individual prin magistrala serială I<sup>2</sup>C permițând utilizarea mai multor module în paralel. Până în anul 2015 plăcuțele Arduino oficiale au folosit cipuri Atmel din seria megaAVR, în special ATmega8, ATmega168, ATmega328, ATmega1280 și ATmega2560, iar în 2015 au fost adăugate cipuri de la alți producători. O multitudine de alte procesoare au fost folosite de dispozitive compatibile Arduino. Multe plăcuțe includ un regulator liniar de 5 V și un oscilator cu cuarț de 16 MHz (sau un rezonator ceramic în unele variante), deși anumite plăcuțe, cum ar fi LilyPad, funcționează la 8 MHz și nu necesită regulator, datorită restricțiilor de formă. Un microcontroler instalat pe Arduino vine preprogramat cu un bootloader care simplifică încărcarea programelor pe memoria flash a cipului, în comparație cu alte dispozitive care necesită programatoare externe. Acest aspect face Arduino o soluție simplă, permițând programarea de pe orice computer ordinar. În prezent, bootloader-ul optiboot este bootloader-ul implicit instalat pe Arduino UNO.

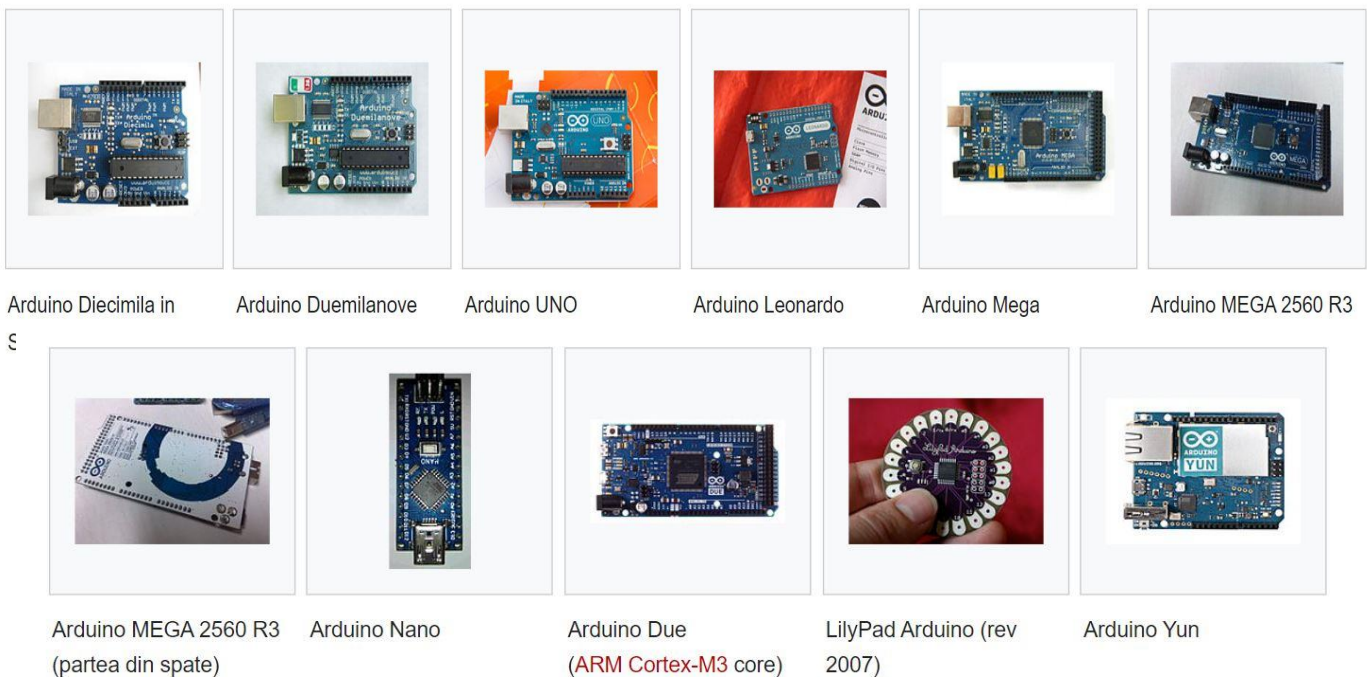
La nivel conceptual, când se folosește mediul de dezvoltare integrat Arduino, programarea tuturor plăcuțelor se face prin conexiune serială. Implementarea acestuia diferă în funcție de versiunea hardware. Unele plăcuțe Arduino au implementate convertoare de nivel logic pentru a

realiza conversia între nivelele logice RS-232 și cele TTL. Plăcuțele Arduino din prezent sunt programate prin USB, având integrate cipuri de conversie USB-serial, cum ar fi FTDI FT232. Unele modele UNO, mai noi, folosesc un cip AVR separat programat să funcționeze ca un convertor USB-serial, care poate fi reprogramat printr-un port ICSP dedicat. Alte variante, cum ar fi Arduino Mini și versiunea neoficială Boarduino, folosesc adaptoare detașabile USB-serial, cabluri, Bluetooth sau alte metode.

Plăcuța Arduino are expuși mulți dintre pinii de intrare/ieșire ai microcontrolerului, pentru ca aceștia să fie folosiți de alte circuite. Diecimila, Duemilanove și UNO oferă 14 pini digitali de intrare/ieșire, dintre care 6 pot produce semnale PWM și 6 intrări analogice care, de asemenea, pot fi folosite ca intrări/ieșiri digitale. Acești pini sunt accesibili prin partea superioară a plăcuței, prin intermediul unor barete mamă cu pasul între pini de 2,54 mm.

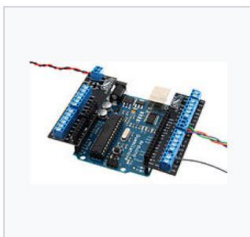
## I.4 Plăcuțe oficiale Arduino + Shield-uri

Plăcuțele originale Arduino erau produse de compania italiană Smart Projects. O parte dintre plăcuțele cu brandul Arduino au fost proiectate de companiile americane SparkFun Electronics și Adafruit Industries. 16 versiuni de hardware Arduino au fost produse în scop comercial până la această dată.





Mai multe shield-uri pot fi suprapuse. În acest exemplu, shield-ul superior conține un breadboard.



Shield cu terminale cu șuruburi dispuse sub formă de aripi.



Shield Adafruit Motor cu terminale cu șurub pentru conectarea motoarelor.



Shield Adafruit pentru înregistrarea datelor cu slot pentru card SD și cip ceas RTC.



Shield HackARobot Fabric - proiectat pentru Arduino Nano, cu posibilitatea de a conecta motoare și senzori, ca un giroscop sau un GPS și alte module cum ar fi: WiFi, Bluetooth, RF, etc.

## I.5 Software

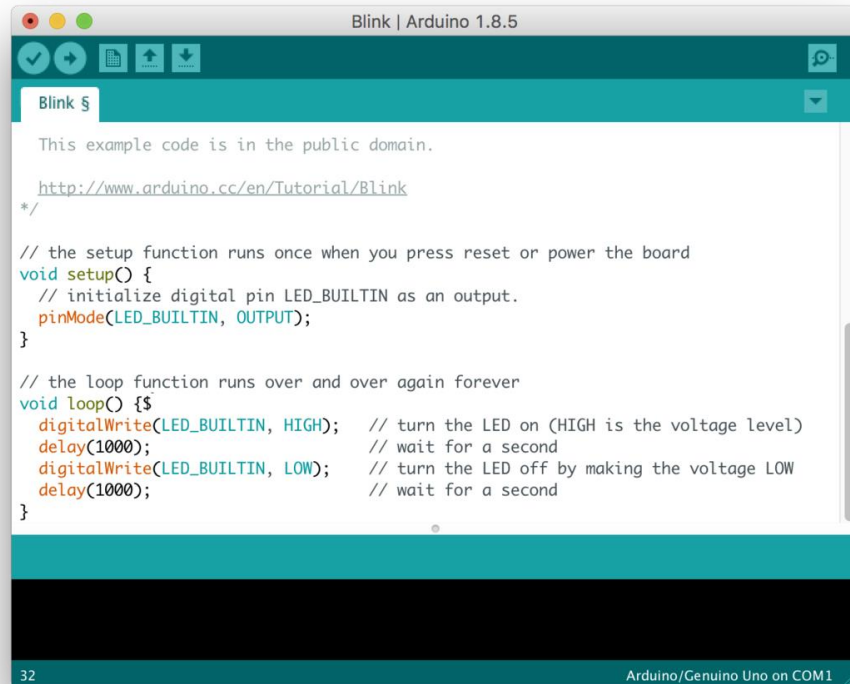
Programele Arduino pot fi scrise în orice limbaj de programare cu un compilator capabil să producă un cod mașină binar. Atmel oferă un mediu de dezvoltare pentru microcontrolerele sale, AVR Studio și mai nou, Atmel Studio.

Proiectul Arduino oferă un mediu integrat de dezvoltare (IDE), care este o aplicație cross-platform, scrisă în Java. Acesta își are originile în mediul de dezvoltare pentru limbajul de programare Processing și în proiectul Wiring. Este proiectat pentru a introduce programarea în lumea artiștilor și a celor nefamiliarizați cu dezvoltarea software. Include un editor de cod cu funcții ca evidențierea sintaxelor, potrivirea acoladelor și spațierea automată și oferă mecanisme simple cu un singur click, pentru a compila și a încărca programele în plăcuța Arduino. Un program scris în IDE pentru Arduino se numește sketch.

Arduino IDE suportă limbajele de programare C și C++ folosind reguli speciale de organizare a codului. Arduino IDE oferă o librărie software numită Wiring, din proiectul Wiring, care oferă multe proceduri comune de intrare și ieșire. Un sketch tipic Arduino scris în C/C++ este compus din două funcții care sunt compilate și legate cu un ciot de program `main()`, într-un program executabil cu o execuție ciclică:

- `setup()`: o funcție care este rulată o singură dată la începutul programului, când se inițializează setările.
- `loop()`: o funcție apelată în mod repetat până la oprirea alimentării cu energie a plăcuței.

După compilarea și legarea cu GNU toolchain inclus, de asemenea, în IDE, mediul de dezvoltare Arduino trimite comandă către programul avrdude pentru a converti codul executabil într-un fișier text codat hexazecimal, care poate fi încărcat în placa Arduino de un program de încărcare.



The screenshot shows the Arduino IDE window titled "Blink | Arduino 1.8.5". The code editor contains the following text:

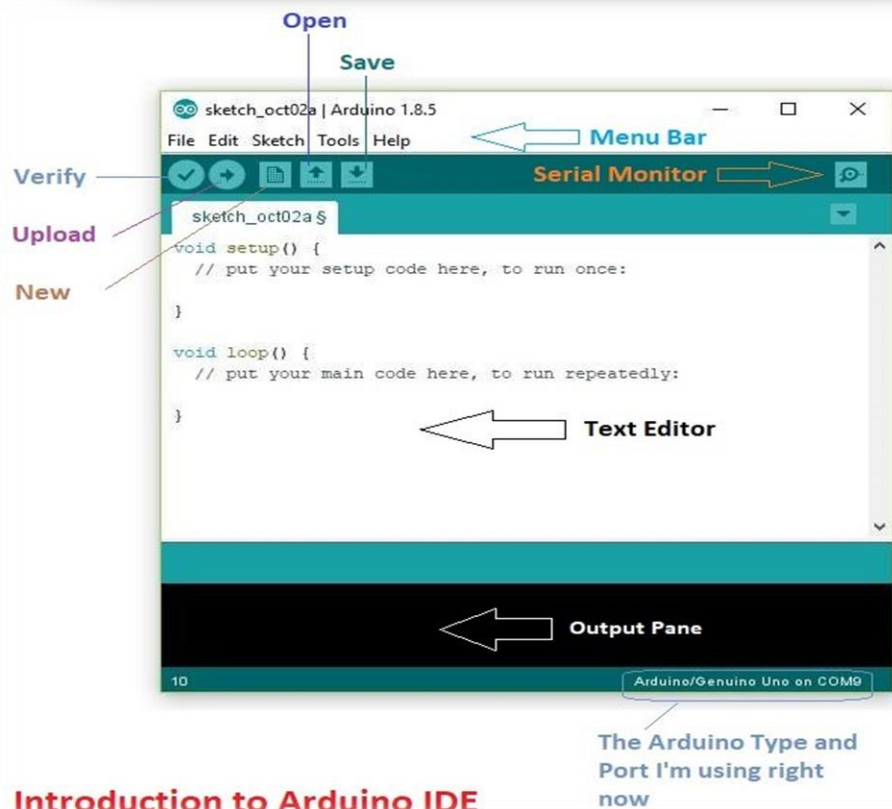
```
This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {$
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}
```

At the bottom of the window, the status bar indicates "32" and "Arduino/Genuino Uno on COM1".



## **Capitolul II – Site Web**

### **II.1 – Limbaje folosite**

#### **II.1.1 – HTML**

HTML este o formă de marcare orientată către prezentarea documentelor text pe o singura pagină, utilizând un software de redare specializat, numit agent utilizator HTML, cel mai bun exemplu de astfel de software fiind browserul web. HTML furnizează mijloacele prin care conținutul un'ui document poate fi adnotat cu diverse tipuri de metadata și indicații de redare. Indicațiile de redare pot varia de la decorațiuni minore ale textului, cum ar fi specificarea faptului că un anumit cuvânt trebuie subliniat sau că o imagine trebuie introdusă, până la scripturi sofisticate, hărți de imagini și formulare. Metadatale pot include informații despre titlul și autorul documentului, informații structurale despre cum este împărțit documentul în diferite segmente, paragrafe, liste, titluri etc. și informații cruciale care permit ca documentul să poată fi legat de alte documente pentru a forma astfel hyperlink-uri (sau web-ul).

HTML este un format text proiectat pentru a putea fi citit și editat de oameni utilizând un editor de text simplu. Totuși scrierea și modificarea paginilor în acest fel solicită cunoștințe solide de HTML și este consumatoare de timp. Editoarele grafice (de tip WYSIWYG) cum ar fi Macromedia Dreamweaver, Adobe GoLive sau Microsoft FrontPage permit ca paginile web să fie tratate asemănător cu documentele Word, dar cu observația că aceste programe generează un cod HTML care este de multe ori de proastă calitate.

HTML se poate genera direct utilizând tehnologii de codare din partea serverului cum ar fi PHP, JSP sau ASP. Multe aplicații ca sistemele de gestionare a conținutului, wiki-uri și forumuri web generează pagini HTML.

#### **II.1.2 – CSS**

CSS sau Cascading Style Sheets este un standard pentru formatarea elementelor unui document HTML. Stilurile se pot atașa elementelor HTML prin intermediul unor fișiere externe sau în cadrul documentului, prin elementul `<style>` și/sau atributul `style`. CSS se poate utiliza și



pentru formatarea elementelor XHTML, XML și SVGL.[1] CSS este unul dintre tehnologiile de bază utilizate în procesul de dezvoltare web, împreună cu HTML și JavaScript.

CSS permite separarea și prezentarea vizuală a conținutului unei pagini web, inclusiv culorile și fonturile disponibile. Separarea elementelor unei pagini îmbunătățește accesibilitatea paginii și permite o mai bună flexibilitate și un control în specificațiile caracteristicilor de prezentare.

### **II.1.3 – JavaScript**

Cea mai des întâlnită utilizare a JavaScript este în scriptarea paginilor web. Programatorii web pot îngloba în paginile HTML script-uri pentru diverse activități cum ar fi verificarea datelor introduse de utilizatori sau crearea de meniuri și alte efecte animate.

Browserele rețin în memorie o reprezentare a unei pagini web sub forma unui arbore de obiecte și pun la dispoziție aceste obiecte script-urilor JavaScript, care le pot citi și manipula. Arborele de obiecte poartă numele de Document Object Model sau DOM. Există un standard W3C pentru DOM-ul pe care trebuie să îl pună la dispoziție un browser, ceea ce oferă premiza scrierii de script-uri portabile, care să funcționeze pe toate browserele. În practică, însă, standardul W3C pentru DOM este incomplet implementat. Deși tendința browserelor este de a se alinia standardului W3C, unele din acestea încă prezintă incompatibilități majore, cum este cazul Internet Explorer.

O tehnică de construire a paginilor web tot mai întâlnită în ultimul timp este AJAX, abreviere de la „Asynchronous JavaScript and XML”. Această tehnică constă în executarea de cereri HTTP în fundal, fără a reîncărca toată pagina web, și actualizarea numai anumitor porțiuni ale paginii prin manipularea DOM-ului paginii. Tehnica AJAX permite construirea unor interfețe web cu timp de răspuns mic, întrucât operația (costisitoare ca timp) de încărcare a unei pagini HTML complete este în mare parte eliminată. JavaScript este cu adevărat un limbaj dinamic. Rar este necesar să utilizați operatorul de evaluare pentru lucruri de genul acesta, pentru că, practic, puteți scrie orice doriți, dacă sintaxa este corectă și dacă ceea ce ați scris nu există, veți primi o eroare de execuție.

## II.2 – Arhitectura Site-ului WEB

Pagina principală (Acasă) – informații despre funcționarea proiectului:

calendar evenimente cu Arduino UNO R3			
Acasa	Componente	Cod	Pagina 4
<p><b>Ideea Proiectului:</b></p> <p>Proiectul are rolul de a face posibila retinerea unor evenimente si afisarea acestora in functie de data la care au fost programate!</p> <p>Retinerea evenimentelor se face conform unui format de tipul ZZ/LL/AAAA -- 00:MM -- TEXT (zi/luna/an -- ora:minut -- TEXT(50c)).</p> <p>Afisarea evenimentelor se face in doua moduri:</p> <ol style="list-style-type: none"> <li>1. Afisand manual toate evenimentele dintr-o zi anume.</li> <li>2. Afisare automata timp de 1 minut daca evenimentul a fost programat la data cand se face afisarea.</li> </ol> <p>Atat introducerea cat si stergerea evenimentelor se realizeaza manual din soft-ul Arduino, prin intermediul Serial-Monitor.</p> <p>Proiectul poate retine pana la 10 evenimente!</p>			
Mandrea Radu Alexandru XII A			

A doua pagină (Componente) – informații despre componentele folosite în proiectul practic:

calendar evenimente cu Arduino UNO R3			
Acasa	Componente	Cod	Pagina 4
<div>Componente:</div> <div>In realizarea proiectului sunt necesare:</div> <div><div>1. Arduino Uno R3</div><div>2. BreadBoard</div><div>3. LCD (Liquid Crystal Display)</div><div>4. RTC (Real Time Clock)</div><div>5. PUSH BUTTONS</div><div>6. JUMPER WIRES</div></div> <div>Aici voi scrie din nou despre componentele folosite in proiectul meu plus o lista cu componentele plus schema electrica daca mai e nevoie.</div>	Arduino Uno R3		
	BreadBoard		
	LCD		
	RTC (Real Time Clock)		
	PUSH BUTTONS		
	JUMPER WIRES		

Mandrea Radu Alexandru XII A

A treia pagină (Cod) – informații privind algoritmi folosiți pentru a programa placa Arduino:

**calendar evenimente cu Arduino UNO R3**

Acasa	Componente	Cod	Pagina 4
-------	------------	-----	----------

**Limbaj de programare folosit:**

Pentru a crea calendarul cu evenimente am folosit limbajul de programare Arduino, limbaj care are la baza arhitectura C++.

In ceea ce priveste site-ul web am folosit:

- HTML
- CSS
- JavaScript

Mai jos voi explica cateva secvente importante din codul Arduino:

<pre>#include "RTClib.h" #include &lt;LiquidCrystal.h&gt; #include &lt;Wire.h&gt; #include &lt;EEPROM.h&gt;</pre>	Bibliotecile folosite.
---	------------------------

A patra pagină (Poze / Video) – conține imagini si videoclipuri de pe parcursul asamblării proiectului.

**calendar evenimente cu Arduino UNO R3**

Acasa	Componente	Cod	Poze / Video
-------	------------	-----	--------------

**POZE + VIDEO MONTAJ PE BREADBOARD**







ALTE POZE MISTO:

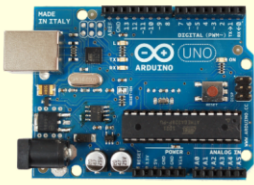







A doua pagină (Componente) conține un submeniu de tip dropdown ce face legătura cu mai multe pagini, câte una pentru fiecare componentă electronică din proiect.

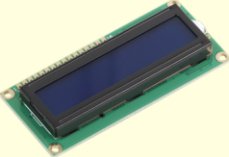
Prima pagină din meniul dropdown conține informații tehnice despre placa de dezvoltare Arduino Uno R3.

calendar evenimente cu Arduino UNO R3			
Acasa	Componente	Cod	Pagina 4
<h3>Placa de dezvoltare Arduino Uno R3</h3> <div style="display: flex; align-items: flex-start;">  <div> <p>Arduino Uno R3 este o placă de dezvoltare bazată pe microcontrollerul ATmega328.</p> <p>Are 14 pini de intrare/ieșire (dintre care 6 pot fi folosiți ca ieșiri PWM), 6 intrări analog, un oscilator de 16MHz, o conexiune USB, mufă de alimentare, și un buton de reset.</p> <p>Poate fi alimentat direct de la calculator, de la portul USB, prin intermediul unei baterii de 9V sau a unui alimentator de 9V.</p> <p>Caracteristici tehnice:</p> <ol style="list-style-type: none"> <li>1. Microcontroller ATmega328</li> <li>2. Tensiune de operare: 5V</li> <li>3. Tensiune de alimentare recomandată: 7-12V</li> <li>4. Limită de tensiune: 6-20V</li> <li>5. Pini intrare/ieșire digitali: 14 (dintre care 6 pot oferi ieșire PWM)</li> <li>6. Pini analogici de intrare: 6</li> <li>7. Memorie Flash 32 KB</li> <li>8. SRAM 2 KB</li> <li>9. EEPROM 1 KB</li> <li>10. Frecvență de lucru: 16 MHz</li> </ol> </div> </div>			

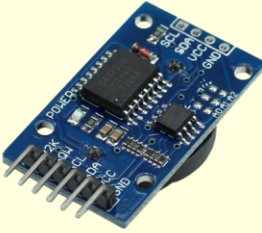
A doua pagină din meniul dropdown conține informații tehnice despre placa de expansiune Breadboard, care facilitează testarea rapidă și ușoară privind funcționalitatea componentelor.

calendar evenimente cu Arduino UNO R3			
Acasa	Componente	Cod	Pagina 4
<h3>Placa de expansiune - BreadBoard</h3> <div style="display: flex; align-items: flex-start;">  <div> <p>Breadboard-ul este folosit în general pentru realizarea rapidă a montajelor fără a fi nevoie de lipirea firelor, pentru testarea proiectelor.</p> <p>Piese se pot conecta prin fire de tip tata-tata, mama-mama sau direct în gaurile din placa breadboard.</p> <p>Fiecare pin se poate conecta prin cele 4 găuri așezate perpendicular pe circuit.</p> <p>Doi magistrale amplasate în lateral, se folosesc în mod normal pentru alimentare, placa având în total, 4 alimentări independente.</p> </div> </div>			



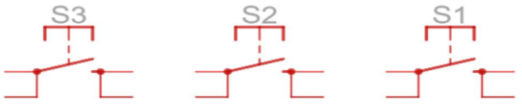
A treia pagină din meniul dropdown conține informații tehnice despre liquid cristal display LCD, care facilitează afișarea rapidă și intuitivă a instrucțiunilor, dar și a informațiilor ce urmează a fi afișate pe ecran și nu pe Serial Monitor.

Calendar evenimente cu Arduino UNO R3			
Acasa	Componente	Cod	Pagina 4
<h3>Liquid Cristal Display - LCD</h3> <div style="display: flex; align-items: flex-start;">  <div> <p><b>Caracteristici tehnice:</b></p> <ol style="list-style-type: none"> <li>1. Tensiune de alimentare: 5V;</li> <li>2. Curent: 1.1mA;</li> <li>3. Tensiune de alimentare backlight: 4.2V;</li> <li>4. Curent backlight: 100mA.</li> </ol> <p><b>Dimensiuni:</b> 80mm x 36mm x 12mm.</p> <p>Modulul este ideal pentru proiecte inovative de electronică ce trebuie să afișeze informații către exterior.</p> <p>LCD-ul este foarte bun și pe întuneric, având iluminare de fundal albastră.</p> <p>Caracterele sunt de culoare albă.</p> <p>LCD-ul are contrast ajustabil, culoarea caracterelor este alba, iar backlight-ul albastru.</p> <p>Comanda LCD-ului este compatibilă cu circuitul integrat HD44780.</p> </div> </div>			

Al patrulea link din meniul dropdown conține informații tehnice despre real time clock, denumit pe scurt RTC. Acest modul ce se montează ușor în orice proiect arduino face posibilă reținerea timpului, dar și a datei fără a fi conectat la o sursă de curent de pe placă, datorită bateriei CR2032.

Calendar evenimente cu Arduino UNO R3			
Acasa	Componente	Cod	Pagina 4
<h3>Real Time Clock - Modul RTC DS1302</h3> <div style="display: flex; align-items: flex-start;">  <div> <p>Modulul este un ceas de precizie în toată regula, cu calendar încorporat.</p> <p>Bateria funcționează ca backup în cazul întreruperii alimentării și poate menține timpul cel puțin un an.</p> <p>Precizia se menține foarte bine și la schimbări de temperatură.</p> <p>Comunicarea cu placa de dezvoltare se face prin interfața I2C.</p> <p><b>Caracteristici tehnice:</b></p> <ol style="list-style-type: none"> <li>1. Tensiune alimentare: 3.3-5.5VDC</li> <li>2. Chip ceas: DS1302</li> <li>3. Chip memorie: AT24C32 cu memorie 32K</li> <li>4. Interfața comunicare: I2C, IIC</li> <li>5. Baterie backup: CR2032 sau LIR2032 (reincarcabilă)</li> <li>6. Senzor temperatura încorporat</li> <li>7. Output alarma programabil</li> </ol> </div> </div>			

Al cincilea link din meniul dropdown conține informații tehnice despre butoanele de tip push. Ulterior aceste butoane au fost schimbate cu niste switch-uri de la o tastatură, pentru a oferi un aspect mai interesant și atractiv proiectului.

Calendar evenimente cu Arduino UNO R3			
Acasa	Componente	Cod	Pagina 4
<b>Push Buttons</b>			
			
			

© Andrei Paul Alexandru 2021

Al șaselea link din meniul dropdown conține informații despre fire conductoare și câte tipuri dintre acestea există. Firele asigură de asemenea testarea rapidă a unor componente electronice, care ulterior vor fi lipite definitiv cu ciocanul de lipit.

Calendar evenimente cu Arduino UNO R3			
Acasa	Componente	Cod	Pagina 4
<b>Jumper Wires - Fire Conductoare</b>			
 <p>Setul de Jumper Wires ajuta in a elimina dezordinea de pe BreadBoard. Acesta vine in diferite lungimi și culori pentru a vă menține circuitul pe suprafață și pentru a fi ușor de identificat.</p> <p>Firele conductoare pot fi de 3 tipuri:</p> <ul style="list-style-type: none"> <li>• MAMA - TATA (folosit pentru a prelungi un singur cablu)</li> <li>• MAMA - MAMA (folosit pentru a prelungi 2 cabluri)</li> <li>• TATA - TATA (pentru a conecta doua elemente din circuit)</li> </ul>			
			

## **Capitolul III – Proiectul practic Calendar Evenimente**

### **III.1 Ce face acest calendar ?**

Proiectul are rolul de a face posibilă reținerea unor evenimente și afișarea acestora în funcție de data la care au fost programate.

Reținerea evenimentelor se face conform unui format de tipul ZZ/LL/AAAA -- OO:MM -- TEXT (zi/lună/an -- oră:minut -- TEXT(50c)).

Afișarea evenimentelor se face în două moduri:

1. Afișând manual toate evenimentele dintr-o zi anume.
2. Afișare automată timp de 1 minut dacă evenimentul a fost programat la data când se face afișarea.

Atât introducerea cât și ștergerea evenimentelor se realizează manual din soft-ul Arduino, prin intermediul Serial-Monitor.

Proiectul poate reține până la 10 evenimente.

### **III.2 Cum funcționează acest calendar ?**

Proiectul este divizat în două părți:

1. Primul mod de funcționare a proiectului nu implică conectarea la calculator (fără a fi deschis portul serial din soft-ul de dezvoltare Arduino), ci doar conectarea la o sursă de curent, de exemplu o baterie sau un acumulator de 5V sau 9V sau o baterie externă. În această ipostază, proiectul are rolul de afișa evenimentele din ziua respectivă.
2. Cel de-al doilea mod de funcționare a proiectului presupune conectarea plăcii de dezvoltare Arduino la un calculator ce are instalat Soft-ul Arduino. Este necesară această conexiune, deoarece avem nevoie de Serial Monitor, dar și implicit de o tastatură, pentru a introduce / șterge evenimente.



### III.2.1 Afișarea evenimentelor din ziua respectivă

Pentru a afla dacă există evenimente în ziua în care se verifică acest lucru, utilizatorul trebuie să acționeze cel de-al treilea buton de la stânga la dreapta printr-o singură apăsare a acestuia. În urma acționării butonului, pe LCD vom observa dacă există evenimente în ziua respectivă. În caz contrar, vom primi un mesaj de informare că nu există evenimente disponibile.





### III.2.2 Introducerea unui eveniment

Pentru a introduce un eveniment, trebuie să lansăm aplicația Arduino, din care ulterior vom deschide Serial Monitor. În acest caz, placa se va conecta prin port-ul serial cu calculatorul, fapt ce ne introduce în cel de-al doilea mod de funcționare a proiectului.

Mai departe, în ceea ce privește introducerea unui nou eveniment, trebuie acționat primul buton, de la stânga la dreapta, printr-o singură apăsare. Odată apăsat, pe LCD vom fi atenționați de faptul că trebuie să introducem un nou eveniment de la tastatură. Introducerea evenimentului are loc doar în urma respectării unui anumit format.

- Dacă formatul introdus de utilizator conține date eronate privind timpul sau data desfășurării evenimentului, acesta va fi atenționat că formatul este greșit, și poate încerca din nou să introducă un eveniment.
- Dacă formatul a fost introdus corect, utilizatorul va observa pe LCD că un eveniment nou a fost introdus, dar va și putea vizualiza data și ora la care a fost introdus, împreună cu un mesaj referitor la eveniment în sine.







### III.2.3 Ștergerea unui eveniment

Pentru a șterge un eveniment, trebuie să lansăm aplicația Arduino, din care ulterior vom deschide Serial Monitor. În acest caz, placa se va conecta prin port-ul serial cu calculatorul, fapt ce ne introduce în cel de-al doilea mod de funcționare a proiectului.

Mai departe, în ceea ce privește ștergerea unui eveniment deja introdus, trebuie acționat cel de-al doilea buton, de la stânga la dreapta, printr-o singură apăsare. Odată apăsate, pe LCD vom fi atenționați de faptul că trebuie să ștergem un eveniment de la tastatură. Ștergerea evenimentului are loc doar în urma respectării unui anumit format, ce implică introducerea numărului de ordine al evenimentului.

- Dacă numărul introdus de utilizator este greșit, adică nu există un eveniment cu acest număr de ordine, sau numărul evenimentelor este mai mic, utilizatorul va primi un mesaj pe LCD care îl atenționează că nu există acest eveniment, dar și afișează numărul evenimentelor.
- Dacă numărul introdus a fost corect, utilizatorul va observa pe LCD că evenimentul cu numărul de ordine introdus anterior a fost șters.





### III.2.4 Afișarea tuturor evenimentelor introduse indiferent de dată

Pentru a afișa toate evenimentele indiferent de data acestora, trebuie să lansăm aplicația Arduino, din care ulterior vom deschide Serial Monitor. În acest caz, placa se va conecta prin port-ul serial cu calculatorul, fapt ce ne introduce în cel de-al treilea mod de funcționare a proiectului.

Mai departe, în ceea ce privește afișarea evenimentelor introduse, trebuie acționat cel de-al treilea buton, de la stânga la dreapta, printr-o singură apăsare. Odată apăsat, proiectul se desprinde din nou în două ramuri precum la cazul **III.2.1**.

1. Dacă există evenimente introduse, mai întâi pe LCD va fi afișat numărul de evenimente introduse, iar apoi timp de câte patru secunde va fi afișat câte un eveniment, până la afișarea tuturor.
2. În caz contrar, dacă în aplicație nu a fost introdus niciun eveniment, sau dacă au fost șterse toate evenimentele, pe LCD va fi afișat un mesaj de atenționare ce explică faptul că nu există evenimente introduse.



## **Surse:**

<https://ro.wikipedia.org/wiki/Arduino>

<https://en.wikipedia.org/wiki/HTML>

<https://ro.wikipedia.org/wiki/JavaScript>

<https://www.tinkercad.com/>

<https://www.arduino.cc/>



## Cuprins:

<b>Introducere .....</b>	<b>1</b>
<b>Capitolul I - Arduino .....</b>	<b>2</b>
I.1 Despre Arduino.....	3
I.2 Istoric Arduino.....	3
I.3 Hardware.....	3
I.4 Plăcuțe oficiale Arduino + Shield-uri .....	4
I.5 Software.....	5
<b>Capitolul II - Site Web.....</b>	<b>7</b>
II.1 Limbaje folosite.....	7
II.1.1 HTML .....	7
II.1.2 CSS.....	7
II.1.3 JavaScript .....	8
II.2 Arhitectura site-ului web.....	9
<b>Capitolul III – Proiectul practic Calendar evenimente.....</b>	<b>14</b>
III.1 Ce face acest calendar? .....	14
III.2 Cum funcționează acest calendar?.....	15
III.2.1 Afișarea evenimentelor din ziua respectivă .....	15
III.2.2 Introducerea unui eveniment .....	16
III.2.3 Ștergerea unui eveniment .....	18
III.2.4 Afișarea tuturor evenimentelor introduse indiferent de dată.....	20