***Colegiul Național "A.T. Laurian" Botoșani***

***Specializarea matematică-informatică***

***Lucrare de atestat profesională la disciplina informatică***

***Profesor coordonator: Pricope Liliana-Mihaela***

***Nume Candidat: Timofti Vlăduț-Constantin***

***Mai 2023***

***Tema lucrării: Portofolio Personal***

**Cuprins**

[**1 - Motivul alegerii temei** 4](#_Toc134555996)

[**2 - Structura aplicației** 5](#_Toc134555997)

[**3 - Limbajul HTML** 7](#_Toc134555998)

[Elementele Limbajului HTML 9](#_Toc134555999)

[Marcaje de bază 10](#_Toc134556000)

[Marcaje pentru structurarea documentului 11](#_Toc134556001)

[Marcaje pentru formatarea textului si crearea listelor 12](#_Toc134556002)

[Marcaje pentru crearea hiperlegăturilor 16](#_Toc134556003)

[Marcaje pentru introducerea de obiecte 17](#_Toc134556004)

[**Exemplu de tabel într-o pagină HTML** 18](#_Toc134556005)

[**4 - Softul utilizat** 22](#_Toc134556006)

[**5 - Resurse de Hard și Soft** 24](#_Toc134556007)

[**6 - Modul de utilizare** 25](#_Toc134556008)

[**7 - Modul de realizare** 26](#_Toc134556009)

[**8 – Bibliografie și webografie** 27](#_Toc134556010)

# **1 - Motivul alegerii temei**

Lucrarea de față are ca temă un portofoliu persoanl. Din meniul principal putem alege:

* Cine sunt eu
* Pasiuni și hobby-uri
* Concursuri și olimpiade
* Gânduri de licean

Aplicația a fost realizată în cea mai mare parte în limbajul HTML, prin programare direct în cod sursă (scrierea liniilor de comandă).

Am ales această temă deoarece îmi doresc ca pe viitor să pot privi în urmă și să văd câte lucruri am reușit să realizez că și copil, încât atunci când simt că totul merge rău, să pot să îmi amintesc cine sunt.

# **2 - Structura aplicației**

Structura lucrării folosește o cromatică verde, îmbinând mai multe nuanțe, realizând în acest mod un contrast care nu deranjează ochiul, permițând cititorului să petreacă mai mult timp fără a avea probleme.

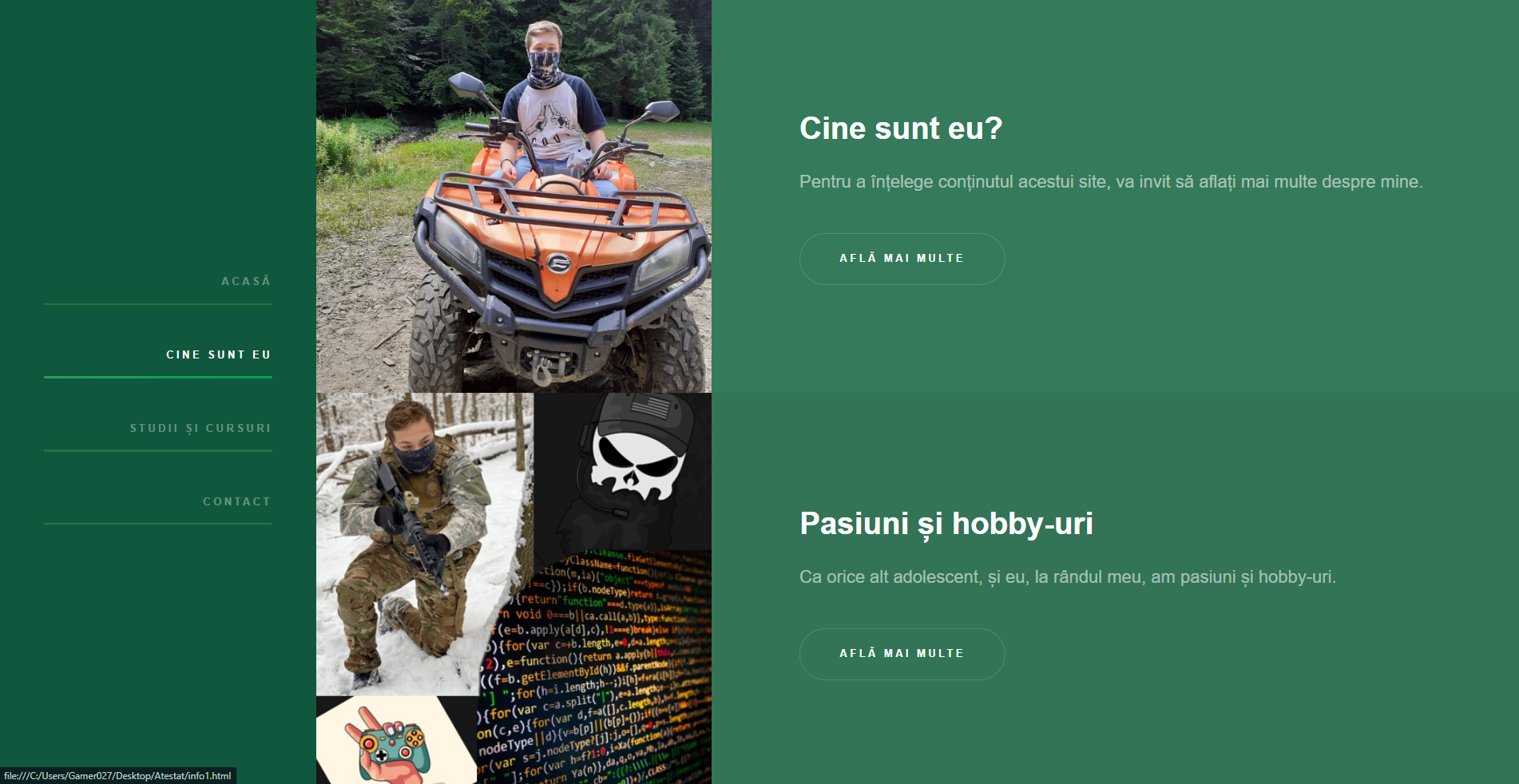
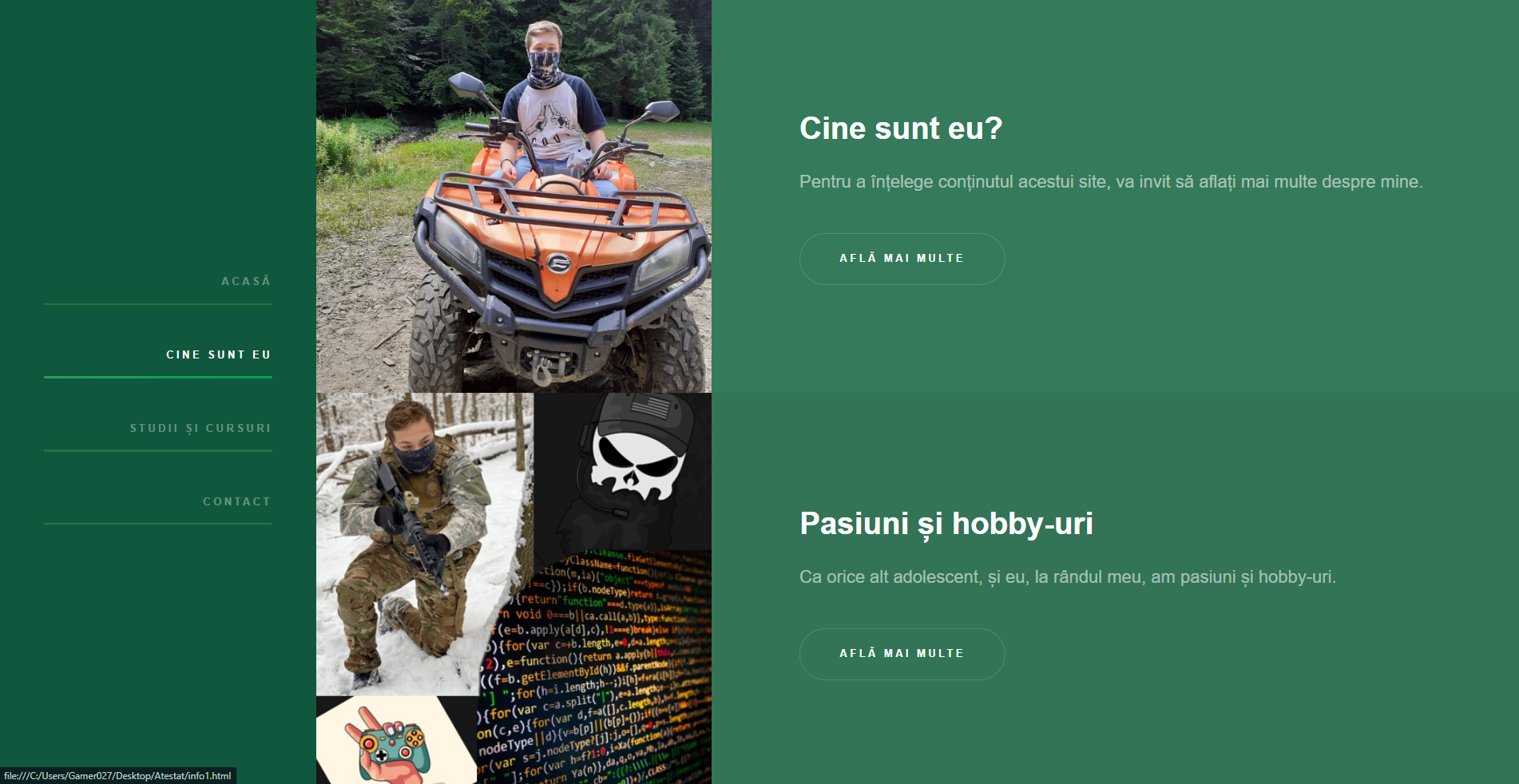
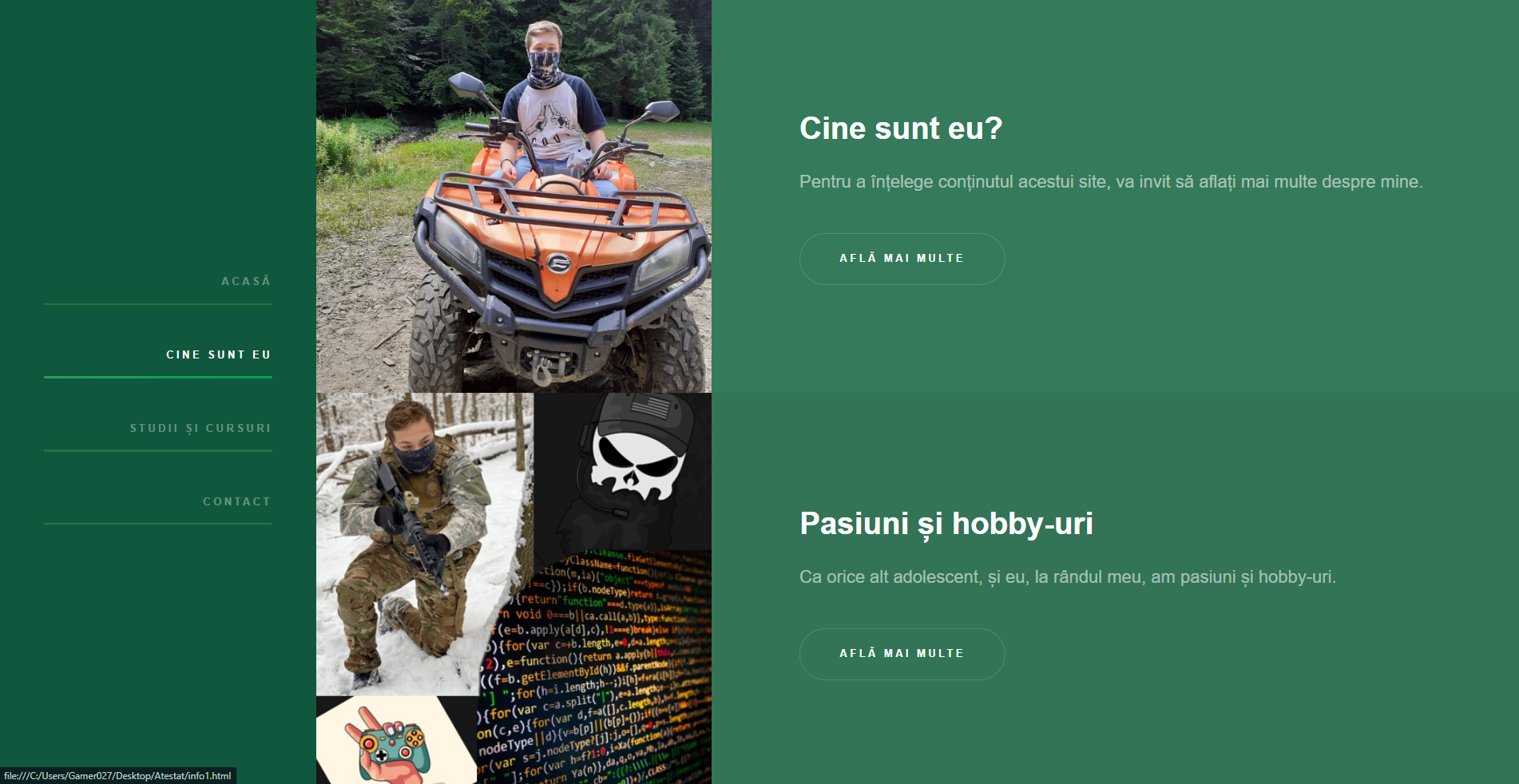
Pagina principală permite trecerea spre diferite link-uri ce reprezint generalități despre mine, cât și studii în domeniul informaticii și al calculatoarelor, oferind astfel informații diverse.

Primul capitol, ***Cine sunt eu?*** mă reprezintă. Aici veți afla mai multe despre mine ca și persoana, începând din perioada grădiniței până în momentul liceului.

Pagina ***Pasiuni și hobby-uri*** prezintă sumar câteva din pasiunile mele principale, alături de hobby-uri dobândite pe parcursul celor 19 ani.

Pagina ***Concursuri și olimpiade*** oferă o idee asupra concursurilor scoloare și extrașcolare la care am participat individual și alături de Palatul Copiilor Botoșani.

Pagina ***Gânduri de licean*** prezintă viziunea mea asupra viitorului, atât pe plan personal cât și profesional.

# **3 - Limbajul HTML**

HTML (Hypertext Markup Language) este un limbaj creat în scopul de a descrie, în mod text, formatul paginilor Web; fisierele create în acest limbaj vor fi interpretate de navigatoare, care vor afisa paginile în forma dorită (cu texte formatate, liste, tabele, formule, imagini, hiperlegături, obiecte multimedia etc.). HTML a apărut ca o aplicatie ISO standard (apartine standardului SGML - Standard Generalized Markup Language, specializat pentru hipertext si adaptat la Web).

Asa cum se poate deduce din numele limbajului, HTML descrie caracteristicile de format ale elementelor incluse prin procedee de marcare. Acestea pot fi asemănate intuitiv cu marcajele folosite în tipografie pentru a indica scrierea unui text cu un anumit tip de caractere. Fiecare element va fi introdus între două marcaje ("tags", în limba engleză) - de început si sfârsit - (uzual) de forma <marcaj> … </marcaj>. Caracterele speciale de delimitare a marcajelor "<", ">" permit deosebirea acestora de textul propriu-zis. De exemplu, pentru textele aldine (îngrosate), marcajul de început este <B> iar de sfârsit - </B>.

În informatică, limbajele de marcare sunt foarte convenabile fiindcă comenzile lor pot fi interpretate simplu. LaTeX-ul, de exemplu, este tot un limbaj de marcare; prin interpretarea fisierelor .tex descrise în acest limbaj se va genera formatul dorit al documentelor pe diverse tipuri de sisteme de calcul (în cazul, LaTeX-ului, se obtine uzual format PostScript sau PDF). În schimb, procesoarele de documente uzuale nu au un limbaj de marcare standardizat, care să ofere compatibilitate între diverse tipuri de calculatoare si sisteme de operare. Astfel, se poate spune că avantajele aplicării limbajelor de marcare constau în portabilitate si flexibilitate: fisierele create cu ajutorul lor pot fi transferate pe orice tip de sistem, unde vor fi interpretate cu ajutorul unor programe specifice.

De fapt, procesoarele de texte uzuale folosesc adesea procedee de marcare pentru formatări (de exemplu, formatările de tip caracter din Word); în acest caz însă, caracterele de control introduse sunt ascunse iar rezultatul editării este direct vizibil ("What You See Is What You Get"). În schimb, în limbajele de marcare - inclusiv HTML - marcajele sunt introduse în text, astfel încât acestea sunt exclusiv succesiuni de caractere (litere, cifre, caractere speciale) - fisiere de tip text.

Referitor la legătura dintre procesoarele de documente uzuale si limbajul HTML, mai trebuie mentionat faptul că ultimele versiuni ale editoarelor de documente oferă facilităti de salvare în format HTML - de exemplu, Word, începând cu versiunea Microsoft Office '97. Mai mult, toate produsele incluse în această gamă dedicată biroticii (MS Office) oferă compatibilitate cu formatul HTML.

Procesele de standardizare si de includere a comenzilor de marcare în fisierele HTML permit navigatoarelor să citească si să formateze paginile Web, lucru foarte important în conditiile în care ele contin nu numai texte alb-negru, ci si culori, imagini, hiperlegături, diverse obiecte. Practic, marcajele HTML asigură controlul asupra modului de afisare a obiectelor corespunzătoare în cadrul programelor de vizualizare a documentelor HTML - navigatoarele.

Limbajul HTML a evoluat în versiuni succesive, odată cu evolutia protocolului HTTP si a programelor de navigare. Astfel, HTML 1.0 era compatibil cu Mosaic, primul program de navigare, dar după aparitia unor navigatoare noi, a fost necesară introducerea unui standard oficial Internet pentru construirea paginilor (HTML 2.0) si extinderea sa cu noi facilităti: formule matematice, tabele, moduri avansate de descriere a organizării paginilor (începând cu HTML 3.0).

Standardizarea oficială a limbajului HTML a fost realizată de consortiul WWW si dezvoltată de diversi producători de soft (unii dintre acestia urmăresc chiar promovarea navigatoarelor proprii prin introducerea unor particularităti în formatele oficiale).

Paginile HTML se pot crea cu orice editor de texte de către utilizatorii care cunosc limbajul HTML sau, mai simplu, se pot utiliza editoare speciale, în care obiectele se introduc interactiv iar codul HTML se generează automat. Având în vedere că si în acest caz este utilă cunoasterea marcajelor generate pentru corectarea eventualelor erori (mai ales în cazul link-urilor), vom prezenta în continuare entitătile care se pot introduce în paginile HTML si marcajele caracteristice acestora:

## =-= Elementele Limbajului HTML =-=

Toate obiectele HTML sunt introduse între marcaje care le definesc; majoritatea acestora sunt de forma <tip\_obiect> (la început) si </tip\_obiect> (la sfârsit). Tipul standard al obiectului poate fi specificat cu majuscule sau minuscule; totusi, se recomandă utilizarea majusculelor fiindcă astfel marcajele ies în evidentă.

Majoritatea navigatoarelor permit vizualizarea paginii curente în formatul sursă HTML (forma pe care o interpretează pentru afisarea paginii). La interpretare, programele de navigare ignoră spatiile si <Enter>-urile, aplicând formatarea specificată.

Există însă si marcaje cu parametri; acestea au forma <tip\_obiect param1=valoare1 param2=valoare2 …>. De exemplu, obiectele de tip imagine sunt introduse cu delimitatorul <IMG>, care are diversi parametri. Pentru definirea hiperlegăturilor se foloseste marcajul <A>, care are de asemenea parametri proprii.  
Codurile de marcare HTML pot fi clasificate în următoarele categorii:

1. marcaje de bază - cele care delimitează pagina / documentul HTML, titlul acesteia si corpul paginii;
2. marcaje pentru structurarea documentului - care permit introducerea de subtitluri, paragrafe, linii de delimitare;
3. marcaje pentru formatarea textului si crearea listelor;
4. marcaje pentru crearea hiperlegăturilor (hyperlinks);
5. marcaje pentru introducerea de obiecte - tabele, formule, imagini sau obiecte multimedia preluate din fisiere, formulare.

Vom descrie în paragrafele următoare, elementele caracteristice fiecăreia din aceste categorii.

Pentru structurarea si organizarea informatiilor din paginile web se pot utiliza frame-uri (marcajul <FRAME>), prin care la un moment dat se afisează mai multe ferestre continând fiecare câte o pagină. Introducerea si gestiunea frame-urilor se realizează foarte convenabil folosind editoarele HTML.

Mai mentionăm faptul că în ultimele versiuni ale limbajului HTML si ale browser-elor s-a introdus posibilitatea integrării, respectiv lansării în executie prin navigator, a unor aplicatii. Acestea sunt scrise în limbajul Java, un limbaj cu caracteristici distribuite si obiectuale, adaptat programării în Web; ele se numesc "applet"-uri si se introduc în sursele HTML cu marcajul <APP> sau <APPLET> . Ultimele versiuni de editoare HTML permit introducerea interactivă a applet-urilor Java.

## =-= Marcaje de bază =-=

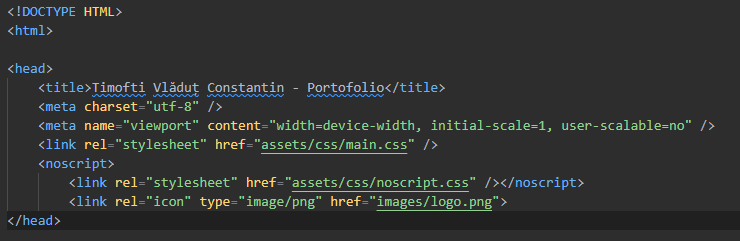
O pagină Web este delimitată de marcajele <HTML> si </HTML> care indică începutul si finalul documentului si contine:

* zonă de antet cuprinsă între marcajele:     <HEAD> … </HEAD> si
* un corp delimitat de marcajele:             <BODY> … </BODY>.

Zona de antet este utilizată de către programele de căutare pe site-urile web si permite specificarea titlului paginii care va fi afisat de navigator (nu apare propriu-zis în continutul paginii).

titlul, introdus în antet, este cuprins între marcajele <TITLE> … </TITLE>.

Astfel, o formă extrem de simplă a unui document HTML ar putea fi cea din următorul exemplu:

<HTML>  
<HEAD> <TITLE>Titlul documentului </TITLE>  
</HEAD>  
<BODY>Document HTML foarte simplu</BODY>  
</HTML>

## =-= Marcaje pentru structurarea documentului =-=

Programele de navigare asigură afisarea diferentiată a unot titluri si subtitluri pentru sectiunile paginii, după criteriile implementate în acest scop la conceperea sa (litere mai mari sau mai mici, diverse culori, litere aldine sau simple, unul sau mai multe rânduri libere după titlu etc.).

Titlurile de capitole sau subtitlurile sunt definite de marcajele <Hn>, unde n este o cifră între 1 si 6 care specifică nivelul titlului (1 este titlul principal iar 6 corespunde celui mai scăzut nivel). Astfel:  
<H1> … </H1>    indică un subtitlu de nivelul 1  
…  
<H6> … </H6>    indică un subtitlu de nivelul 6.

Pentru separarea zonelor paginii se pot folosi treceri la:

* linie nouă - cu marcajul <BR>, eventual cu desenarea unei linii orizontale - marcajul <HR>;
* paragraf nou - cu marcajul <P> (se inserează o linie nouă si eventual se face o indentare). Marcajul </P> desemnează sfârsitul de paragraf dar este mai rar folosit (se poate omite).

## =-= Marcaje pentru formatarea textului si crearea listelor =-=

Formatările uzuale de texte permit scrierea cu caractere:

* aldine - în acest scop pentru acel text se aplică marcajele     <B>… </B>;
* cursive (italice) - pentru acel text se aplică marcajele    <I> … </I>;
* subliniate - textul se introduce între marcajele    <U> …</U>.

Unele programe mai vechi de navigare nu permiteau reprezentarea acestor formate; în acest caz, se aplicau alte moduri de evidentiere a textelor respective (culori, video invers).

Alte tipuri de formatări de caractere care se pot defini se referă la:

dimensiunea fonturilor si culori - se va utiliza marcajul cu parametri:

<FONT SIZE=x COLOR=y> … </FONT> , care indică utilizarea unui font de dimensiune x si culoare y.

* x poate fi un număr între 1 si 12, indicând mărimea exactă, sau un număr cu semn, indicând mărimea relativă la dimensiunea curentă.
* y poate fi numele unei culori standard, specificată în limba engleză si între ghilimele, sau, eventual, descompunerea unei culori în componentele RGB, sub formă hexazecimală (câte două cifre hexa pentru fiecare componentă).

Suplimentar, se pot crea pagini de stiluri, în care să se definească stiluri logice, modificabile ulterior; în acest scop, se folosesc marcajele <DN> - definitie, <EM> - punere în evidentă, <STRONG> - accentuare puternică.

Pentru tastaturile care suportă numai codurile ASCII, caracterele speciale se pot crea folosind formatul &nume\_caracter - de exemplu, &egrave; pentru e. Având în vedere semnificatia specială a caracterelor <, > si &, aparitia lor în documente trebuie specificată sub forma caracterelor speciale: &lt; , &gt; , respectiv &amp; .

Prezentăm în continuare un exemplu de document HTML, precum si modul în care acesta este afisat de Netscape Navigator.

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML//EN">  
<html>  
  
<head>  
<title>Pagina simpla</title>  
</head>  
  
<body>  
<p>Acesta este un exemplu simplu de o   
pagina HTML. Ea arata cum   
pot aparea caractere <strong>bold</strong>, <em>italic</em>,   
<font size="5">mai mari</font>,   
<font size="1">mai mici</font>,   
respectiv insera   
o mica imagine   
<img src="Handshake8114.gif"  
width="20" height="14">.  
</p>  
</body>  
  
</html>

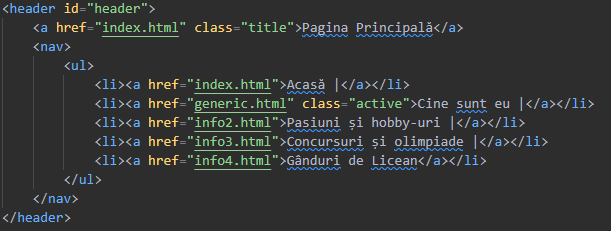
Introducerea obiectelor de tip imagine (marcajul <IMG>) este detaliată mai jos.

Limbajul HTML permite definirea mai multor tipuri de liste si imbricarea lor (includerea unor liste în altele), caz în care trebuie verificată cu atentie corespondenta dintre marcajele de început si sfârsit pentru fiecare listă. Formatarea listelor la afisarea paginii (introducerea bulinelor, numerotării etc.) se face de către programul de navigare.

Elementele listelor se introduc între marcajele <LI> … </LI> ("list item"). Modul de aparitie al elementelor depinde de tipul de listă în care sunt incluse (cu buline, numerotate etc.), determinat de marcajul specific.

Astfel, listele pot fi:

* neordonate (cu buline) - delimitate de marcajele <UL> … </UL> ("unordered list"); elementele lor se afisează cu buline;
* ordonate (numerotate) - delimitate de marcajele <OL> … </OL> ("ordered list"); elementele lor se vor numerota;
* de tip meniu - delimitate de marcajele <MENU> … </MENU>; elementele acestor liste apar într-o reprezentare compactă pe ecran (multicoloană);
* glosare (liste de definitii) delimitate de marcajele <DL> … </DL> ("definition list"); elementele acestora au câte două componente, introduse respectiv cu marcajele <DT> (pentru nume) si <DD> (pentru semnificatia asociată);
* tabele scurte - create cu marcajul <DIR>.

Listele utilizate cel mai frecvent sunt cele ordonate si neordonate. Prezentăm în continuare un exemplu simplu de pagină HTML care contine aceste tipuri de liste.  
  
<html>  
  
<head>  
<title>Pagina cu liste</title>  
</head>  
  
<body>  
  
<p>Aceasta pagina exemplifica crearea listelor</p>  
  
<p>Cele mai uzual folosite tipuri de liste sunt:   
  
<ul>  
<li>neordonate</li>  
<li>ordonate</li>  
</ul>  
  
<p>O lista ordonata este o lista numerotata:   
  
<ol>  
<li>primul element;</li>  
<li>al doilea element...</li>  
</ol>  
  
</body>  
</html>

## =-= Marcaje pentru crearea hiperlegăturilor =-=

Hiperlegăturile ("hyperlinks") se introduc cu marcajele <A> … </A> ("anchor") aplicate asupra unui text sau a unei imagini. Ele se pot crea către:

* adrese URL - astfel se asigură accesul în cadrul procesului de navigare (prin protocoale specifice), nu numai la paginile web, ci si la diverse servicii Internet;
* fisiere / resurse locale adică aflate pe acelasi calculator cu pagina creată - link-uri locale ; acestea ar putea fi accesate si printr-un URL cu protocolul "file" dar pentru mai multă simplitate, este suficientă specificarea numelui si căii fisierului local;
* o zonă din documentul HTML curent - link-uri interne.

În fiecare din aceste cazuri, un click pe textul sau imaginea respectivă va determina activarea legăturii si afisarea prin intermediul navigatorului a resursei asociate link-ului.

Resursa asociată unei legături este descrisă prin parametrii marcajului <A>:

* HREF - permite asocierea unei adresei URL, crearea unei legături locale prin specificarea numelui si căii fisierului corespunzător sau a unei legături interne documentului; este cel mai important parametru pentru construirea de hiperlegături;
* NAME - numele hiperlegăturii sau al zonei dintr-o pagină către care se face referirea, în cazul unui link intern;
* METHODS - metodele de acces (acest parametru se utilizează doar în prelucrări speciale).  
  Astfel, un link local sau către o adresă URL se poate defini pe un text cu: <A HREF="resursa"> text </A>.

O legătură internă către o zonă a documentului curent se indică folosind marcajul <A NAME="nume-zona">.

## =-= Marcaje pentru introducerea de obiecte =-=

Prima versiune de HTML (1.0) nu permitea descrierea tabelelor sau a informatiilor formatate. Asemenea obiecte trebuiau create formatat si introduse ca obiect deja formatat, cu marcajele <PRE>, </PRE>. Aceste marcaje indicau navigatorului că textul inclus trebuia afisat caracter cu caracter, fără vreo interventie de formatare. În versiunile ulterioare ale limbajului HTML, s-au introdus însă facilităti elegante de formatare, de includere a tabelelor si a altor obiecte.

În HTML 3.0 s-au introdus obiectele de tip tabel, formate din linii orizontale si coloane verticale la a căror intersectie se formează celulele. Acestea pot contine intrări diverse: texte, figuri sau chiar alte tabele. Se pot realiza reuniri de celule (de exemplu, pentru titluri mai lungi). Atributele de formatare a tabelului (aliniere, stilul chenarului si marginilor etc.) se definesc în proiectarea paginii dar modul lor de afisare va depinde si de programul de navigare.

Tabelele se introduc între marcajele <TABLE> … </TABLE>, cărora li se pot atasa (optional) parametrii BORDER si RULES. Un titlu pentru tabel se poate introduce cu marcajul <CAPTION>. Fiecare coloană se defineste cu marcajul <COL>, având ca parametru ALIGN - modul de aliniere a informatiilor din acea coloană (LEFT, CENTER, RIGHT).

Antetul tabelului se poate indica între marcajele <TH>…</TH> ("Table Header"), pentru trecerea la o linie nouă se utilizează marcajul <TR> ("Table Row") iar celulele individuale se marchează cu <TD> ("Table Data"), eventual cu parametru de aliniere. Se mai pot specifica alinieri orizontale sau verticale ale celulelor, grupări de celule etc. Aceste marcaje permit navigatorului să afiseze diferentiat informatiile din tabel.

Prezentăm în continuare un exemplu simplu de tabel.

<HTML>  
<HEAD>  
<TITLE>Pagina cu tabel</TITLE>  
</HEAD>  
<BODY>  
<H1>Aceasta pagina da un exemplu de folosire a unui tabel</H1>  
<H3>Vanzari anuale:</H3>  
<TABLE> <TH>1996</TH><TH>1997</TH><TH>1998</TH><TR>  
<TD>125 milioane lei</TD><TD>160 milioane lei</TD><TD>231 milioane lei</TD>  
</TABLE>  
</BODY>  
</HTML>

### **=-= Exemplu de tabel într-o pagină HTML =-=**

Mărimea liniilor si a coloanelor poate fi controlată suplimentar prin marcajele <ROWSPAN=x> si respectiv <COLSPAN=y>, fiecare indicând mărimea în "celule normale". În plus, se poate indica trasarea unei margini pentru tabel, prin marcajul <TABLE BORDER>.

Includerea imaginilor se face folosind marcajul <IMG>, care are parametri specifici:

* SRC indică (după semnul '=') URL-ul imaginii, respectiv calea fisierului dacă acesta este local; uzual, se acceptă fisiere în format GIF sau JPEG;
* ALLIGN controlează alinierea imaginii fată de limita inferioară a textului (TOP, MIDDLE sau BOTTOM); este un parametru optional;
* ALT furnizează textul afisat în locul imaginii dacă utilizatorul dezactivează optiunea de afisare a imaginilor (parametru optional);
* ISMAP este un indicator optional pentru imaginile care sunt hărti selectabile.

Astfel, introducerea simplă a unei imagini se poate realiza cu <IMG SRC="specificare-imagine">. Un exemplu de imagine inserată într-o pagină HTML este prezentat în figura de mai sus.  
Remarcăm faptul că, utilizând parametri specifici, marcajul <IMG> se poate folosi si pentru inserarea unor fisiere multimedia, cum ar fi secventele video. Având în vedere că aceste operatii se realizează mai usor folsind un editor HTML.

Formulare. HTML 1.0 permitea uzual doar transferarea informatiilor de la furnizori către utilizatori, transferul în sens invers fiind foarte dificil. Odată cu dezvoltarea Web-ului si mai ales cu utilizarea sa în scopuri comerciale si în diverse alte domenii, s-a făcut simtită necesitatea comunicării în dublu sens pentru preluări de comenzi, completarea unor fise de înregistrare, distribuirea de produse soft, administrarea de chestionar, transmiterea unor informatii personale etc. Acestea au fost motivatiile introducerii formularelor în HTML 2.0.

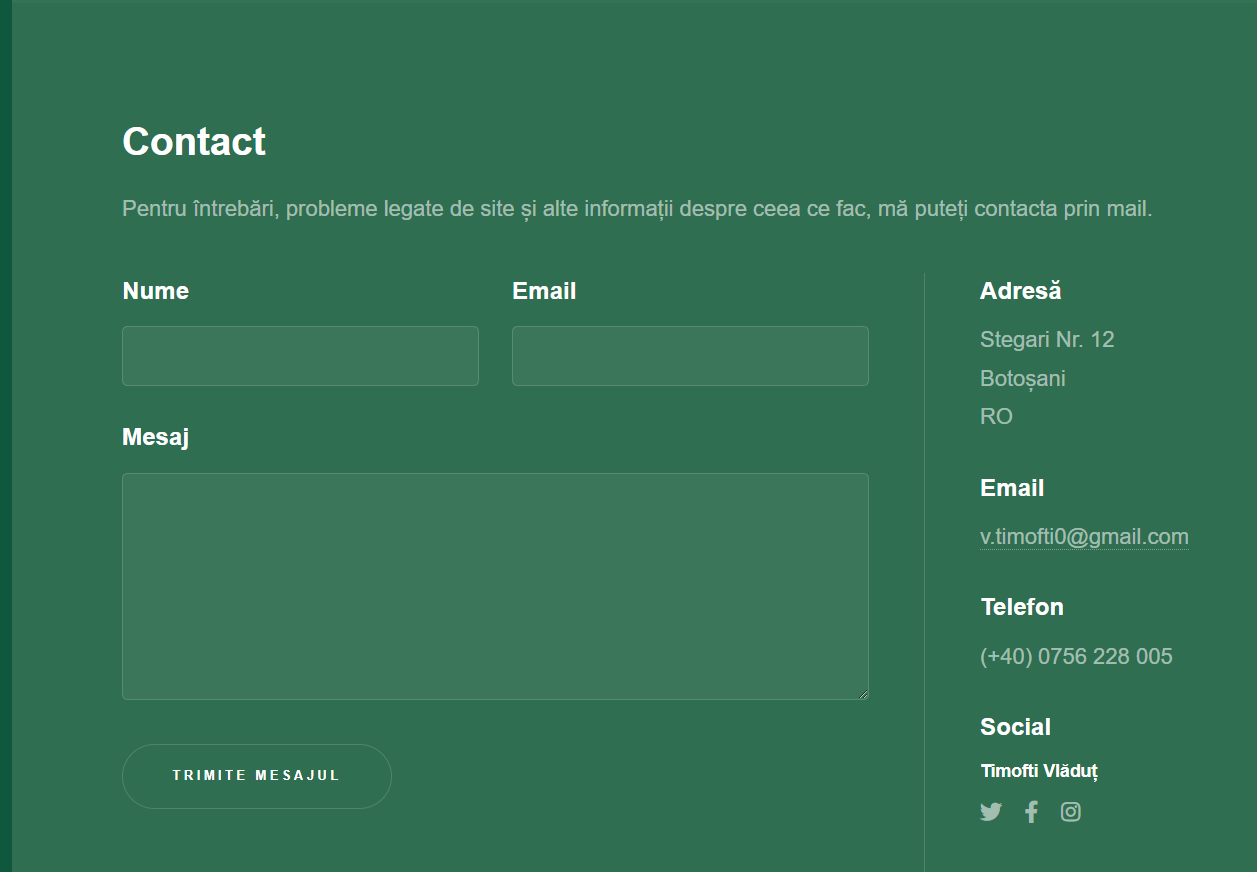
Formularele contin obiecte de control care permit utilizatorilor introducerea de informatii prin completarea unor câmpuri specifice (casete de text), prin selectarea sau activarea unor optiuni (comutatoare, grupuri sau liste de optiuni). Aceste informatii vor fi transmise, după activarea butonului SUBMIT (echivalentă cu tastarea lui <Enter>) proprietarului paginii, introduse în baze de date dedicate si prelucrate cu aplicatii specifice.

Formularele se introduc prin marcajelele <FORM> ... </FORM> , cărora li se atasează parametri specifici care definesc metodele de transmitere si tratare a datelor. Un formular poate contine obiecte de control de diverse tipuri, definite cu marcajul <INPUT> si o varietate de parametri care stabilesc, pentru obiectul definit, tipul, dimensiunea si modul lui de afisare. Astfel, se pot crea câmpuri de text (care vor fi completate cu texte de către utilizator), liste de alternative, comutatoare, grupuri de optiuni, butoane, hărti active etc.

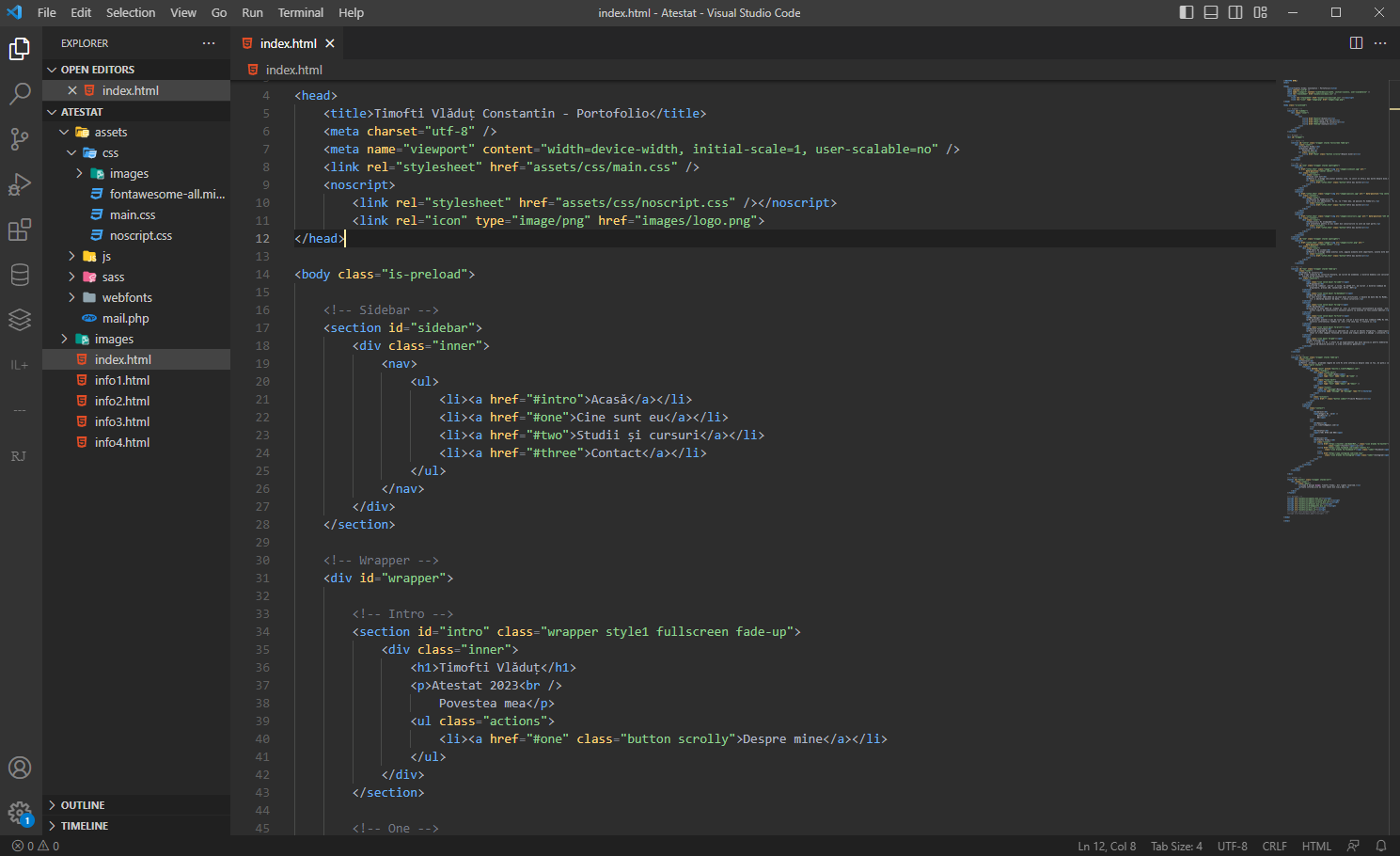
Datele preluate prin intermediul formularelor se tratează conform standardului CGI (Common Gateway Interface) prin intermediul unor programe sau script-uri CGI. În plus, un asemenea script poate interactiona cu baza de date creată pentru realizarea unor actiuni specifice (de exemplu, hărtile active folosesc script-uri CGI pentru a executa diferite actiuni, în functie de zona selectată de utilizator).

Script-urile CGI pot executa si alte operatii decât prelucrarea formularelor, producând iesiri convenabile. Dacă o hiperlegătură indică spre un script CGI, la selectarea legăturii se va executa script-ul (cu anumite variabile de context care retin diverse informatii de stare). Script-ul va produce un fisier, de exemplu o pagină web, care va fi interpretat(ă) de navigator. Acest mecanism permite script-urilor să genereze, aproape instantaneu, pagini web care să satisfacă diverse cerinte ale utilizatorilor, furnizând anumite răspunsuri asteptate în urma unor actiuni.

Crearea interactivă a formularelor, folosind produse soft specializate în realizarea de pagini Web este relativ accesibilă pentru utilizatorii familiarizati cu caracteristicile obiectelor de control specifice aplicatiilor soft dar tratarea ulterioară a informatiilor transmise necesită cunostinte de specialitate.

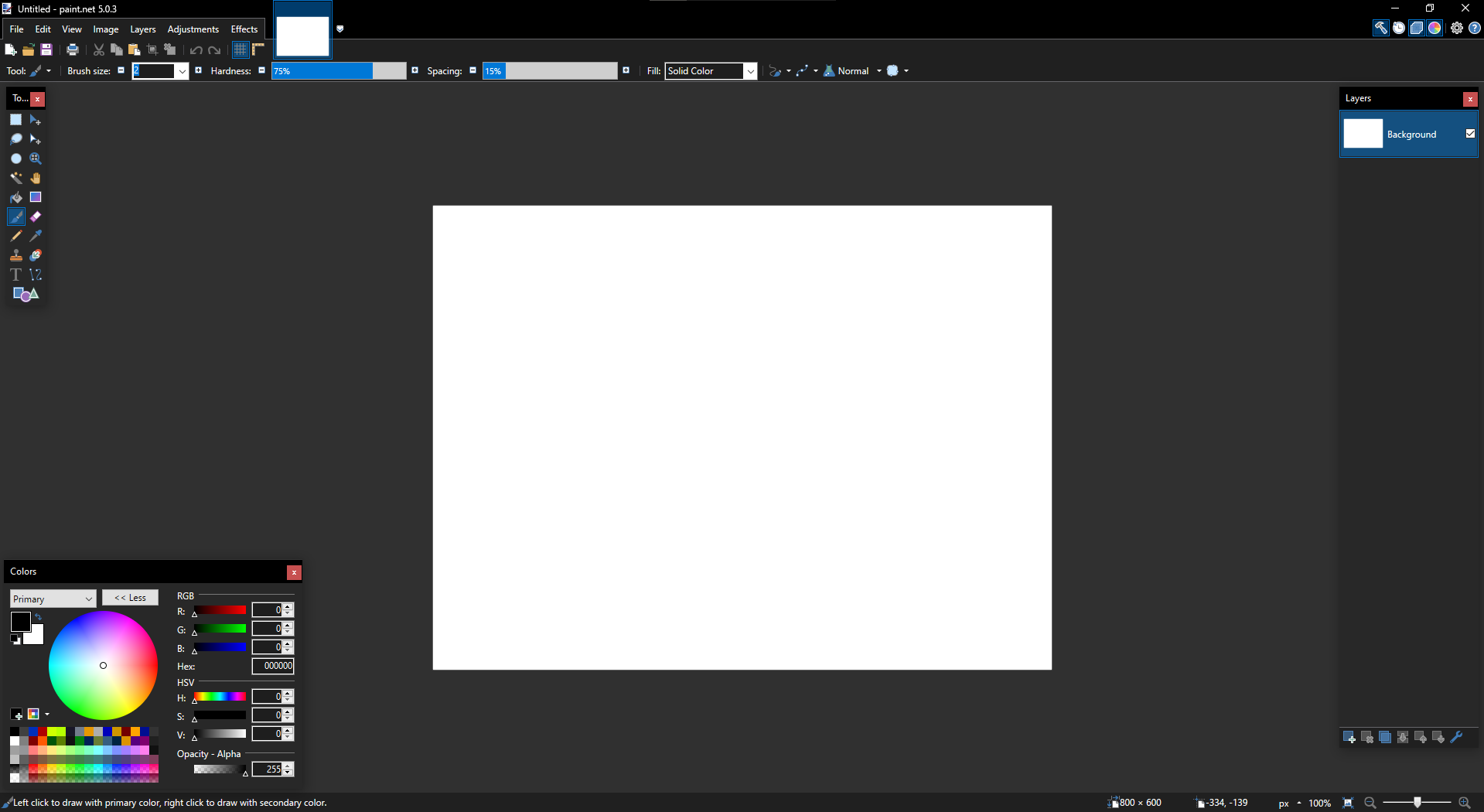
Numele, simbolurile si procedura de introducere interactivă dintr-un editor HTML (de exemplu, Microsoft Front Page) a obiectelor de control sunt extrem de apropiate de cele asociate obiectelor similare din formularele sau rapoartele Microsoft Access. Pentru utilizatorii mai putin ****avizati însă, crearea, chiar interactivă, a formularelor este mai dificilă.

**4 - Softul utilizat**

Pentru a realiza acest atestat, am folosit programul Visual Studio Code (VS Code) alături de mai multe extensii precum:

* **Auto Close Tag;**
* **Auto Rename Tag;**
* **Java Decompiler;**
* **Language Support for Java;**
* **Material Icon Theme;**
* **New Moon Syntax Theme;**
* **Prettier – Code Formatter;**
* **Code Spell Checker;**
* **Live Server;**
* **Open In Browser;**
* **Refsharp;**
* **Serial Monitor.**

Fiecare extensie adițională prezentată mai sus are un rol specific, fie că îmi permite să modific linii de cod în fișiere Java, sau îmi oferă posibilitatea de a aranja într-un mod ușor de citit codul scris.

Pentru realizarea pozelor găsite în pagină, am folosit programul gratuit Paint.NET alături de diverse extensii și plugin-uri care să îmi permit o ușoară manevrare asupra imaginilor editate. Pozele au fost luate din galeria personală și dețin complet drepturi de autor asupra editarilor realizate.

# **5 - Resurse de Hard și Soft**

Tot proiectul a fost realizat pe calculatorul personal, care dispune de următoarele componente:

* **Carcasa Floston RED BARS**
* **Procesor Intel Coffee Lake, Core I3 8100 3.6 GHz**
* **Placa de baza MSI H310M PRO-VD**
* **Susrsa Deepcool Aurora Series DA500**
* **Memorie Corsair Vengeance LPX Black 16GB DDR4 2400 MHz CL 16 dual Channel Kit**
* **Hard Dik WD Blue 1TB SATA-III 7200 RPM 64MB**
* **Placa vide Inno3D GeForce GTX 1660 TWIN X2 OC RGB 6GB GDDR5 192-bit**
* **Tastatura Genesis RHOD 400**
* **Mouse Genesis CX33**

Calculatorul rulează pe **Windows 10 PRO**, varianta de 64-bit.

# **6 - Modul de utilizare**

 Pagina utilizează meta “viewport” ceea ce îi permite să se ajusteze pe fiecare device de pe care este accesat, fie că este telefon, tabletă, laptop sau desktop.

 Este ușor de folosit, întrucât pagina întâmpină utilizatorul cu o informație simplă, numele meu și titlul atestatului, prezentându-i ulterior meniul din partea stângă.

În acest meniu apar punctele principale și de interes pe care utilizatorul le poate dori, și anume “Acasă”, “Cine sunt eu”, “Studii și cursuri” și pagină de contact. De asemenea, pe pagină principala apare și butonul “Despre mine” care în urma apăsării va direcționa utilizatorul folosind o animație, la conținutul atestatului.

După accesarea butonului “Despre mine”, pagina devine foarte simplă, fiecare buton și redirecționare fiind de la sine explicită și ușor de utilizat.

# **7 - Modul de realizare**

Realizarea proiectului a fost sistematica, urmand cativa pasi simpli:

* Alegerea temei – dupa o sesiune lunga de brainstorming in care nu am gasit nicio idee pentru proiect, m-am gandit sa il realizez despre ce stiu cel mai bine, viata mea. Aceasta tema mi-a oferit o gama mai larga de posibilitati de realizare, si o varietate de palete de culori.
* Determinarea paletei de culori – pentru a face o pagina web buna, culorile care o realizeaza trebuie sa nu oboseasca utilizatorul si sa ofere o senzatie de liniste, asa ca am mers cu diferite nuante deschise de verde.
* Realizarea site-ului propriu-zis – acest pas nu a fost neaparat cel mai greu, deoarece site-ul este simplu, prezinta doar cateva animatii, tabele, butoane si un formular. Aici am fost nevoit sa gandesc mai mult paginile care il vor alcatui, sa ma uit prin poze vechi si amintiri incat sa ofer cea mai buna versiune a proiectului pe care o puteam face.
* Editarea foto – fiecare poza in parte a trebuit sa fie modificata, fie cu scris, fie cu ajustari subtile pentru a se incadra bine in pagina.
* Scrierea documentatiei – acest pas este unul fundamental si final, intrucat aici apare fiecare etapa parcursa pentru proiect, apar coduri folosite si multe idei din interior.

# **8 – Bibliografie și webografie**

[*http://www.w3schools.com/*](http://www.w3schools.com/)

[*https://en.wikipedia.org/wiki/Visual\_Studio\_Code*](https://en.wikipedia.org/wiki/Visual_Studio_Code)

[*https://ro.wikipedia.org/wiki/HTML5*](https://ro.wikipedia.org/wiki/HTML5)

[*https://ro.wikipedia.org/wiki/HyperText\_Markup\_Language*](https://ro.wikipedia.org/wiki/HyperText_Markup_Language)

[*https://ro.wikipedia.org/wiki/Cascading\_Style\_Sheets*](https://ro.wikipedia.org/wiki/Cascading_Style_Sheets)

[*https://tutorialehtml.com/ro/introducere-in-html/*](https://tutorialehtml.com/ro/introducere-in-html/)

**Jeremy Keith, Rachel Andrew (2016): HTML5 for Web Designers, Second Edition, 2nd Edition**

**Jennifer Niederst Robbins (2016): HTML5 Pocket Reference, 5th Edition**

**Alex Libby, Gaurav Gupta, Asoj Talesra (2016): Responsive Web Design with HTML5 and CSS3 Essentials**