

# LUCRARE PENTRU ATESTAREA COMPETENȚELOR PROFESIONALE

*Elev :* **Matei Octavian Mihai**

*Profesor îndrumător :* **Niculescu Florin**

*Liceul :* **Colegiul National „Carol I”, Craiova**

*Sesiunea :* **mai 2019**

# Cuprins

Motivul alegerii temei .....	3
Structura Aplicatiei .....	4
Elementele limbajului HTML.....	7
Prezentarea programului in care a fost realizat proiectul .....	18
Resurse de hard și soft necesare.....	20
Instructiuni de utilizare.....	21
Ultime informatii.....	22
Biografie .....	23

## **Motivul alegerii temei**

Tema aleasa reflecta pasiunea mea pentru incercarea intelegerii functionarii lumii fizice si explorarea elementelor esentiale lor, in special atomul si componentele sale. Spre deosebire de celelalte teme posibile, aceasta mi-a permis sa invat si sa explorez o lume spectaculoasa, auzita de multi, dar inteleasa de foarte putini, chiar Niels Bohr, unul dintre fondatorii legilor cunoscute de noi in mecanica cuantica spunea ca "Dacă mecanica cuantică nu te-a șocat profund, încă nu ai înțeles-o."

## Structura Aplicatiei

Site-ul contine mai multe elemente cheie, unele scrise in HTML, CSS, JavaScript, altele scrise in C. Scripturile principale sunt urmatoarele:

- Home.html – pagina principal in care se gasesc intr-un tab de meniu toate componentele atomului si o scurta descriere a proiectului
- Galery.html – pagina in care se afla o lista cu oamenii ce au contribuit la informatiile cunoscute de noi astazi despre atom
- main.js si router.js ce asigura o buna funcitonare a serverului site-ului cu ajutorul tehnologiei Node.js
- Start si Server( programe C) ce asigura un mediu user-friendly in ceea ce priveste pornirea serverului si site-ului

Toate paginile si informatiile generate contin si un link catre pagina de Wikipedia pentru o mai buna eventuala informare a utilizatorului in ceea ce priveste componentele si oamenii de stiinta.

### ***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 afișa paginile în forma dorită (cu texte formate, liste, tabele, formule, imagini, hiperlegături, obiecte multimedia etc.). HTML a apărut ca o aplicație ISO standard (apartine standardului SGML - Standard Generalized Markup Language, specializat pentru hipertext și adaptat la Web).

Așa 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 și sfârșit - (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 sfârșit - `</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 fișierelor .tex descrise în acest limbaj se va genera formatul dorit al documentelor pe diverse tipuri de sisteme de calcul (în cazul, LaTeX-ului, se obține 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 și sisteme de operare. Astfel, se poate spune că avantajele aplicării limbajelor de marcare constau în portabilitate și flexibilitate: fișierele 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 formătări (de exemplu, formătărilor 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) - fișiere de tip text.

Referitor la legătura dintre procesoarele de documente uzuale și limbajul HTML, mai trebuie menționat faptul că ultimele versiuni ale editoarelor de

documente oferă facilități 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 și de includere a comenzilor de marcare în fișierele HTML permit navigatoarelor să citească și să formateze paginile Web, lucru foarte important în condițiile în care ele conțin nu numai texte alb-negru, ci și culori, imagini, hiperlegături, diverse obiecte. Practic, marcajele HTML asigură controlul asupra modului de afișare a obiectelor corespunzătoare în cadrul programelor de vizualizare a documentelor HTML - navigatoarele.

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

Standardizarea oficială a limbajului HTML a fost realizată de consorțiul WWW și dezvoltată de diverși producători de soft (unii dintre aceștia urmăresc chiar promovarea navigatoarelor proprii prin introducerea unor particularități 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 și în acest caz este utilă cunoașterea marcajelor generate pentru corectarea eventualelor erori (mai ales în cazul link-urilor), vom prezenta în continuare entitățile care se pot introduce în paginile HTML și 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) și `</tip_obiect>` (la sfârșit). Tipul standard al obiectului poate fi specificat cu majuscule sau minuscule; totuși, se recomandă utilizarea majusculelor fiindcă astfel marcajele ies în evidență.

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

Există însă și 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 diverși parametri. Pentru definirea hiperlegăturilor se folosește 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 și corpul paginii;
2. marcaje pentru structurarea documentului - care permit introducerea de subtitluri, paragrafe, linii de delimitare;
3. marcaje pentru formatarea textului și crearea listelor;
4. marcaje pentru crearea hiperlegăturilor (hyperlinks);
5. marcaje pentru introducerea de obiecte - tabele, formule, imagini sau obiecte multimedia preluate din fișiere, formulare.

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

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

Mai menționăm faptul că în ultimele versiuni ale limbajului HTML și ale browser-elor s-a introdus posibilitatea integrării, respectiv lansării în execuție prin navigator, a unor aplicații. Acestea sunt scrise în limbajul Java, un limbaj cu caracteristici distribuite și 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 unor 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 înserează o linie nouă și eventual se face o indentare). Marcajul </P> desemnează sfârșitul de paragraf dar este mai rar folosit (se poate omite).

## Marcaje pentru formatarea textului și 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 evidențiere a textelor respective (culori, video invers).

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

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

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

- x poate fi un număr între 1 și 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 și 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> - definiție, <EM> - punere în evidență, <STRONG> - accentuare puternică.

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

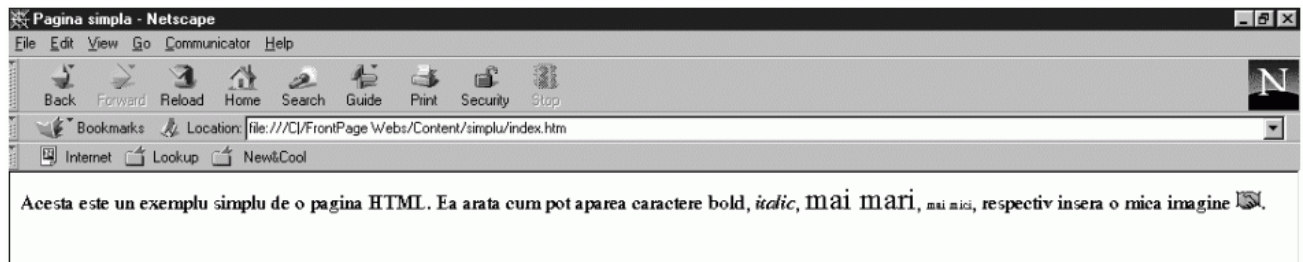
Prezentăm în continuare un exemplu de document HTML, precum și modul în care acesta este afișat de Netscape Navigator.

```
<!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
.
</p>
</body>

</html>
```



În figura, se observă că textul propriu-zis al documentului poate fi delimitat față de codurile de marcare, iar formatarea textului din documentul HTML nu corespunde cu cea afișată de programul de vizualizare:

- textul este extins pentru a ocupa întregul ecran și este afișat cu un corp de literă diferit față de sursa HTML;
- terminatorii de linie nu sunt respectați;

- spațiile albe suplimentare din documentul sursă sunt ignorate.

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

Limbajul HTML permite definirea mai multor tipuri de liste și imbricarea lor (includerea unor liste în altele), caz în care trebuie verificată cu atenție corespondența dintre marcasele de început și sfârșit pentru fiecare listă. Formatarea listelor la afișarea paginii (introducerea bulinelor, numerotării etc.) se face de către programul de navigare.

Elementele listelor se introduc între marcasele <LI> ... </LI> ("list item"). Modul de apariție 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 marcasele <UL> ... </UL> ("unordered list"); elementele lor se afișează cu buline;
- ordonate (numerotate) - delimitate de marcasele <OL> ... </OL> ("ordered list"); elementele lor se vor numerota;
- de tip meniu - delimitate de marcasele <MENU> ... </MENU>; elementele acestor liste apar într-o reprezentare compactă pe ecran (multicoloană);
- glosare (liste de definiții) delimitate de marcasele <DL> ... </DL> ("definition list"); elementele acestora au câte două componente, introduse respectiv cu marcasele <DT> (pentru nume) și <DD> (pentru semnificația asociată);
- tabele scurte - create cu marcajul <DIR>.

Listele utilizate cel mai frecvent sunt cele ordonate și neordonate. Prezentăm în continuare un exemplu simplu de pagină HTML care conține 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>

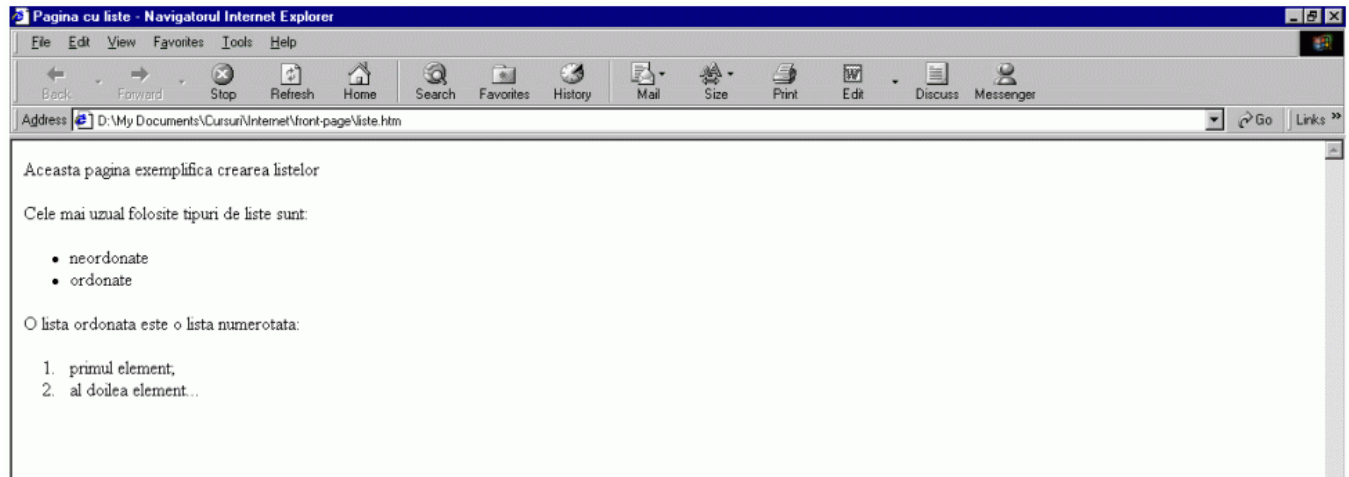
<li>primul element;</li>

<li>al doilea element...</li>

</ol>

</body>

</html>



## *Exemple de liste într-o pagină 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 și la diverse servicii Internet;
- fișiere / resurse locale adică aflate pe același calculator cu pagina creată - link-uri locale ; acestea ar putea fi accesate și printr-un URL cu protocolul "file" dar pentru mai multă simplitate, este suficientă specificarea numelui și căii fișierului 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 și 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 și căii fișierului 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 informațiilor formate. Asemenea obiecte trebuiau create formatat și introduse ca obiect deja formatat, cu marcajele <PRE>, </PRE>. Aceste marcaje indicau

navigadorului că textul inclus trebuia afișat caracter cu caracter, fără vreo intervenție de formatare. În versiunile ulterioare ale limbajului HTML, s-au introdus însă facilități elegante de formatare, de includere a tabelelor și a altor obiecte.

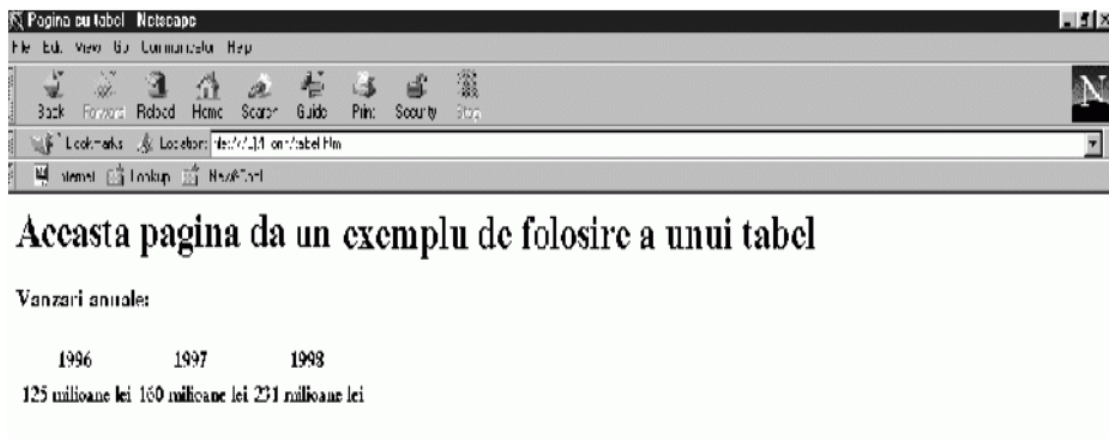
În HTML 3.0 s-au introdus obiectele de tip tabel, formate din linii orizontale și coloane verticale la a căror intersecție se formează celulele. Acestea pot conține intrări diverse: texte, figuri sau chiar alte tabele. Se pot realiza reuniri de celule (de exemplu, pentru titluri mai lungi). Atributele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

Tabelele se introduc între marcajele `<TABLE> ... </TABLE>`, cărora li se pot atașa (optional) parametrii `BORDER` și `RULES`. Un titlu pentru tabel se poate introduce cu marcajul `<CAPTION>`. Fiecare coloană se definește cu marcajul `<COL>`, având ca parametru `ALIGN` - modul de aliniere a informațiilor 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 afișeze diferențiat informațiile 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 și a coloanelor poate fi controlată suplimentar prin marcasele `<ROWSPAN=x>` și 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 fișierului dacă acesta este local; uzual, se acceptă fișiere în format GIF sau JPEG;
- ALIGN controlează alinierea imaginii față de limita inferioară a textului (TOP, MIDDLE sau BOTTOM); este un parametru optional;
- ALT furnizează textul afișat în locul imaginii dacă utilizatorul dezactivează opțiunea de afișare a imaginilor (parametru optional);
- ISMAP este un indicator optional pentru imaginile care sunt hărți 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 și pentru inserarea unor fișiere multimedia, cum ar fi secvențele video. Având în vedere că aceste operații se realizează mai ușor folosind un editor HTML.

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

Formularele conțin obiecte de control care permit utilizatorilor introducerea de informații prin completarea unor câmpuri specifice (casete de text), prin selectarea sau activarea unor opțiuni (comutatoare, grupuri sau liste de opțiuni). Aceste informații vor fi transmise, după activarea butonului SUBMIT (echivalentă cu tastarea lui `<Enter>`) proprietarului paginii, introduse în baze de date dedicate și prelucrate cu aplicații 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, hărți 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, hărțile 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.

# **Prezentarea programului în care a fost realizat proiectul**

## **1. Visual Studio Code**

Visual Studio Code este un editor de cod sursă dezvoltat de Microsoft pentru Windows, Linux și MacOS. Acesta include suport pentru depanare, control Git încorporat și GitHub, evidențierea sintaxei, completarea codului inteligent, fragmente și refactorizarea codului. Este foarte personalizabil, permițând utilizatorilor să schimbe tema, comenzile rapide de la tastatură, preferințele și să instaleze extensii care adaugă funcționalități suplimentare. Codul sursă este gratuit și open source și este disponibil sub licența permisivă MIT. Binarele compilate sunt gratuite și gratuite pentru uz privat sau comercial.

Codul Visual Studio se bazează pe Electron, un cadru care se utilizează pentru a implementa aplicațiile Node.js pentru desktopul care rulează pe motorul de aspect Blink. Deși utilizează cadrul Electron, software-ul nu utilizează Atom și utilizează în același timp aceeași componentă a editorului (codificată "Monaco") utilizată în Azure DevOps (denumită anterior Visual Studio Online și Visual Studio Team Services).

În sondajul de analiză a dezvoltatorilor Stack Overflow 2019, Visual Studio Code a fost clasat cel mai popular instrument de mediu pentru dezvoltatori, 50,7% din 87,317 respondenți pretind că îl folosesc.

## **Istorie**

Codul Visual Studio a fost anunțat în data de 29 aprilie 2015 de către Microsoft la conferința Build 2015. Un preview preview a fost lansat la scurt timp după aceea. Pe 18 noiembrie 2015, Visual Studio Code a fost lansat sub licența MIT și codul său sursă postat la GitHub. De asemenea, a fost anunțat sprijinul pentru extindere. În data de 14 aprilie 2016, Visual Studio Code a absolvit etapa de examinare a publicului și a fost lansată pe web.

## **Caracteristici**

Visual Studio Code este un editor de cod sursă care poate fi folosit cu o varietate de limbi de programare. În loc de un sistem de proiect, acesta permite utilizatorilor să deschidă unul sau mai multe directoare, care apoi pot fi salvate în spații de lucru pentru reutilizare ulterioară. Acest lucru îi permite să funcționeze ca editor de cod lingvistic-agnostic pentru orice limbă, spre deosebire de Microsoft Visual Studio, care utilizează fișierul de soluție .sln proprietate și fișiere de proiect specifice proiectului. Acesta acceptă o serie de limbi de programare și un set de caracteristici care diferă pe fiecare limbă. Fișierele și folderele nedorite pot fi

excluse din arborele proiectului prin intermediul setărilor. Multe dintre caracteristicile Visual Studio Code nu sunt expuse prin meniuri sau interfața cu utilizatorul, dar pot fi accesate prin intermediul paletei de comandă.

Codul Visual Studio poate fi extins prin plug-in-uri, disponibile printr-un depozit central. Aceasta include adăugiri la editor și suport lingvistic. O caracteristică notabilă este capacitatea de a crea extensii care să adauge suport pentru limbi noi, teme, depanare, să efectueze analize de coduri statice, să adauge linteri de cod, să utilizeze protocolul Language Server și să se conecteze la servicii suplimentare.

Codul Visual Studio include mai multe extensii pentru FTP, permițând software-ul să fie folosit ca alternativă gratuită pentru dezvoltarea web. Codul poate fi sincronizat între editor și server, fără a descărca software suplimentar.

Codul Visual Studio permite utilizatorilor să stabilească pagina de cod în care este stocat documentul activ, caracterul de linie nouă pentru Windows / Linux și limba de programare a documentului activ. Aceasta permite utilizarea acestuia pe orice platformă, în orice localizare și pentru orice limbaj de programare dat.

## **Resurse de hard și soft necesare**

### Resurse minime :

#### 1) Hardware :

- *Procesor (CPU) : 500 MHz +*
- *Memorie (RAM): 64 MB*
- *Spațiu liber pe hard-disk : 30 MB*
- *Monitor :1024\*768*
- *Un browser oarecare: Internet explore ,Mozilla, Opera*

#### 2) Software :

- *Sistem de operare : Windows 98/XP, Node.js*

### *Resurse recomandate:*

#### 1) Hardware :

- *Procesor (CPU) : 1000 MHz +*
- *Memorie (RAM): 128 MB*
- *Spațiu liber pe hard-disk : 30 MB*
- *Monitor : VGA 32-bit color (1024x768)*
- *Un browser oarecare: Internet explorer ,Mozilla*

#### 2) Software :

- *Sistem de operare : Windows XP , Node.js*

## **Instructiuni de utilizare**

In cazul in care pe masina pe care se utilizeaza site-ul nu exista Node.js, se va porni programul Server si se va instalata framework-ul.

In cazul in care exista deja Node.js sau s-a instalat cu succes, se porneste programul Start, aplicatie ce va porni serverul codat in main.js si router.js si o pagina Chrome cu ajutorul comenzii “system("start chrome.exe <http://127.0.0.1:3000/home.html>");” si se va utiliza o libraria future din C ca sa se asigure multithreading astfel incat serverul si chrome sa porneasca concomitant.

## Ultime informatii

Paginile sunt generate in mare parte dintr-o functie cu attribute echivalente cu structurile din C, astfel incat se asigura generarea de 8 pagini virtuale intr-un singur URL in home si de afisarea mai multor oameni de stiinta fara a crea o multitudine de nesturi <p> </p>.

## Biografie

- <https://ro.wikipedia.org/wiki/Atom>
- <https://ro.wikipedia.org/wiki/Neutron>
- <https://ro.wikipedia.org/wiki/Proton>
- <https://ro.wikipedia.org/wiki/Electron>
- <https://ro.wikipedia.org/wiki/Quarc>
- <https://www.scientia.ro/fizica/atomul/6117-ce-sunt-leptonii.html>
- <https://ro.wikipedia.org/wiki/Boson>
- [https://ro.wikipedia.org/wiki/Joseph\\_John\\_Thomson](https://ro.wikipedia.org/wiki/Joseph_John_Thomson)
- [https://ro.wikipedia.org/wiki/Ernest\\_Rutherford](https://ro.wikipedia.org/wiki/Ernest_Rutherford)
- [https://ro.wikipedia.org/wiki/Max\\_Planck](https://ro.wikipedia.org/wiki/Max_Planck)
- [https://ro.wikipedia.org/wiki/Niels\\_Bohr](https://ro.wikipedia.org/wiki/Niels_Bohr)
- [https://ro.wikipedia.org/wiki/Werner\\_Heisenberg](https://ro.wikipedia.org/wiki/Werner_Heisenberg)
- [https://ro.wikipedia.org/wiki/Erwin\\_Schrödinger](https://ro.wikipedia.org/wiki/Erwin_Schrödinger)