

## CAP. 1

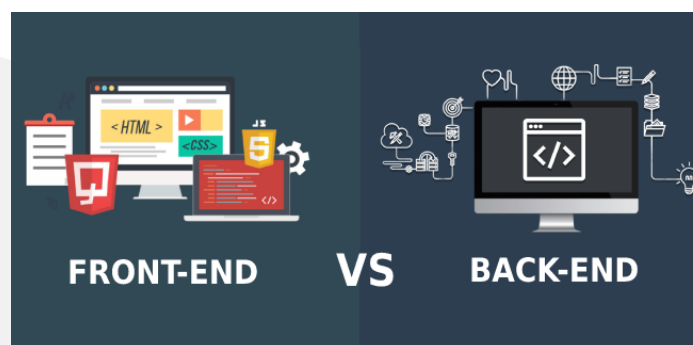
# INTRODUCERE ÎN PROGRAMARE

Internetul este plin de informații, cursuri, tutoriale însă chiar și pentru cineva care are ceva experiență, uneori totul poate deveni confuz, derutant, chiar frustrant, motiv pentru care atunci cineva îți poate da o părere, o cale către a face totul mai ușor, mai pe înțeles, este mai mult decât util. Mai ales, dacă ești la început de drum.

Scopul acestui curs este să facă lucrurile să fie mai simple, mai ușor de înțeles și să te ajute să mergi în direcția corectă.

O aplicație web este formată din:

- **Front End (Client Side)** - presupune designul, structura și animațiile unei aplicații web. Este implementat cu ajutorul limbajelor HTML, CSS și Javascript.
- **Back End (Server Side)** – presupune logica aplicației, partea de server și de baze de date. Este implementat cu ajutorul limbajelor de programare PHP, Python, Java, C# etc. și cu a bazelor de date SQL și NoSQL.



## 1.1 Front End, Back End, Full Stack

Un **Front End Developer** este persoana responsabilă cu implementarea designului unei aplicații web (website, aplicație mobilă, aplicație desktop) cu ajutorul limbajelor HTML, CSS și Javascript. Mai pe scurt, partea pe care clientul o vede și cu care interacționează.

Tot ce vezi pe o pagină web este realizat de un Front End Developer, un fotograf a făcut pozele, un designer a creat logoul și graficile, un copywriter a scris textul, dar Front End Developerul este cel care a asamblat toate aceste piese transformându-le cu ajutorul limbajelor de programare într-o pagină web structurată și interactivă. Pe scurt, un Front End Developer se ocupă de "Client Side".

Deși acesta este un curs de Front End Developer, este important să știm care este rolul unui Back End Developer sau a unui Full Stack Developer.

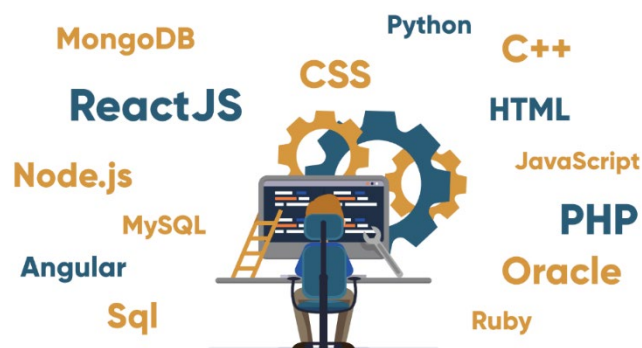
Un **Back End Developer** este persoana responsabilă cu crearea, implementarea și menținerea unei aplicații web funcționale. De fiecare dată când accesăm un website de pe un dispozitiv, serverul ne trimite înapoi niste informații, aceste informații se transformă în pagina web pe care o vedem. De fiecare dată când ne creăm un cont sau ne logăm pe un website, când cumpărăm un produs online sau ne facem un abonament la o platformă de streaming îi putem mulțumi unui



Back End Developer. El este cel care implementează logica unei aplicații, bazele de date în care se găsesc produsele din magazinele online sau conturile noastre de pe anumite website-uri. Pe scurt, un Back End Developer se ocupă de “Server Side”.

Un **Full Stack Developer** este combinația dintre un Front End Developer și un Back End Developer. El este persoana responsabilă cu toată experiența pe care o ai pe un website. El se ocupă de implementarea designului (Client Side) dar și de implementarea părții de server, baze de date (Server Side) și tot el face și legătură dintre “Client Side” și “Server Side”.

Fiecare pagină web are nevoie de o structură definită de către dezvoltator pentru a pune în legătură stilizarea și efectele vizuale, chiar dacă este produsul final a unei aplicații complete, motiv pentru care vom începe direct cu HTML.



## 1.2 Ce este HTML, CSS, JavaScript?

**HTML** (HyperText Markup Language) este un limbaj de marcare folosit pentru crearea structurii unei pagini web, de aceea îi se mai spune scheletul unei pagini web. El are scopul de prezentare a informațiilor (paragrafe, titluri, imagini etc.).

**CSS** (Cascading Style Sheets) permite stilizarea paginii web, a conținutului și aranjarea elementelor în pagină.

**JavaScript** este un limbaj de scripting pe partea de client interpretat de către browser și permite crearea interacțiunilor cu acțiunile userilor pe pagină.

## 1.3 Cum a apărut HTML?

HTML a fost dezvoltat inițial de Tim Berners-Lee în 1989 la CERN. A fost văzut ca o posibilitate pentru oamenii de știință care foloseau computere diferite și voiau să schimbe informații între ei folosind internetul. Pentru asta erau necesare câteva trăsături: independența de platformă, posibilități hypertext și structurarea documentelor.

HTML din prima generație este înțeles de primele navigatoare (modul text). Nivelul 1 este obligatoriu pentru toate navigatoarele și înseamnă posibilitatea de a interpreta text și imagini. Ușor, acesta a început să fie dezvoltat din ce în ce mai mult, după cum urmează:

- **HTML 1.0** și **HTML+** au apărut în 1990, respectiv 1993.
- **HTML 2.0**, apărut în 1994, a fost prima versiune standardizată; ea conține 49 de taguri.
- **HTML 3.0** a apărut în 1995; versiunea cuprindea extensii importante, cum ar fi marcaje pentru notații matematice, bannere etc. În prezent nu mai este utilizat.
- **HTML 3.2**, introdus în 1996, este considerat ca succesorul versiunii 2.0, incorporând o serie de taguri din HTML 3.0 ca și extensii Netscape.

## 1.4 Ce se întâmplă când accesezi o pagină web?

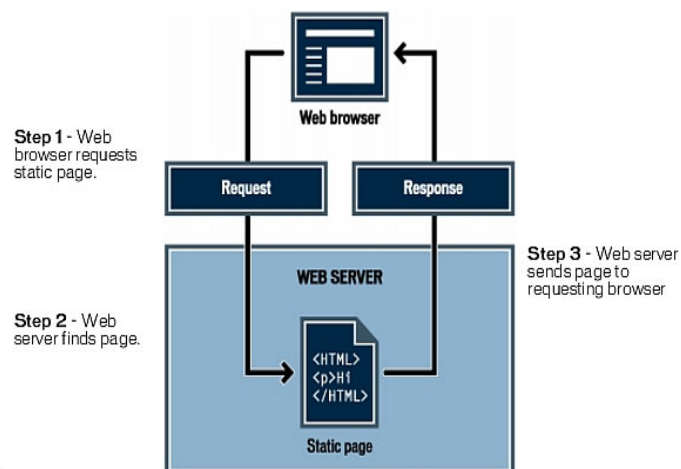
Pentru a înțelege ce se întâmplă când accesăm o pagină web trebuie să înțelegem ce este un server web. Un **server web** poate face referire la o parte **hardware** sau **software**, sau ambele lucrând împreună.

Partea **hardware** presupune un computer unde sunt stocate soft-ul serverului web și componentele websitetului (fișiere html, cs, js). El se conectează la internet și suportă interschimbări de date cu alte dispozitive conectate la internet.

Partea **software** presupune metoda prin care sunt accesate datele de pe partea hardware. La bază, acesta este un server HTTP. Un server HTTP este un software care înțelege adresele URL (adresele web, link-urile) și HTTP este un protocol folosit de browsere care te ajuta să poți accesa și vizualiza paginile web.

În momentul în care vrei să accesezi o pagină, browserul trimite o cerere către serverul web unde sunt stocate fișierele cu ajutorul protocolului HTTP. Când cererea ajunge la serverul web (hardware), serverul HTTP (software) acceptă cererea, găsește fișierele cerute și le trimite înapoi către browser tot prin ajutorul protocolului HTTP, moment în care tie îți este afișată pagina web în browser.

**Observație:** dacă serverul nu găsește fișierele cerute (link-ul este greșit sau șters) acesta trimite eroare 404.



## 1.5 Tool-uri necesare

Pentru acest curs avem nevoie să instalăm următoarele:

- Un editor de Cod (IDE), VsCode
- Un Browser (Chrome sau Firefox)
- Mediu de dezvoltare local Laragon, unealtă universală pentru orice web developer
- Adobe Xd, tool pentru grafică

Pentru a crea un fișier HTML avem la dispoziție 2 metode:

**1. Folosirea unui editor IDE** (Integrated Development Environment) – este un software cu interfață grafică care ne va usura munca prin numeroasele funcții pe care le oferă. Ne ajută să fim mai eficienți prin funcțiile de auto completare a codului, putem vedea mai ușor greșelile și erorile + multe alte avantaje pe care le veți observa pe parcursul cursului.

Avem diferite alternative free și cu plată, cele mai populare fiind:

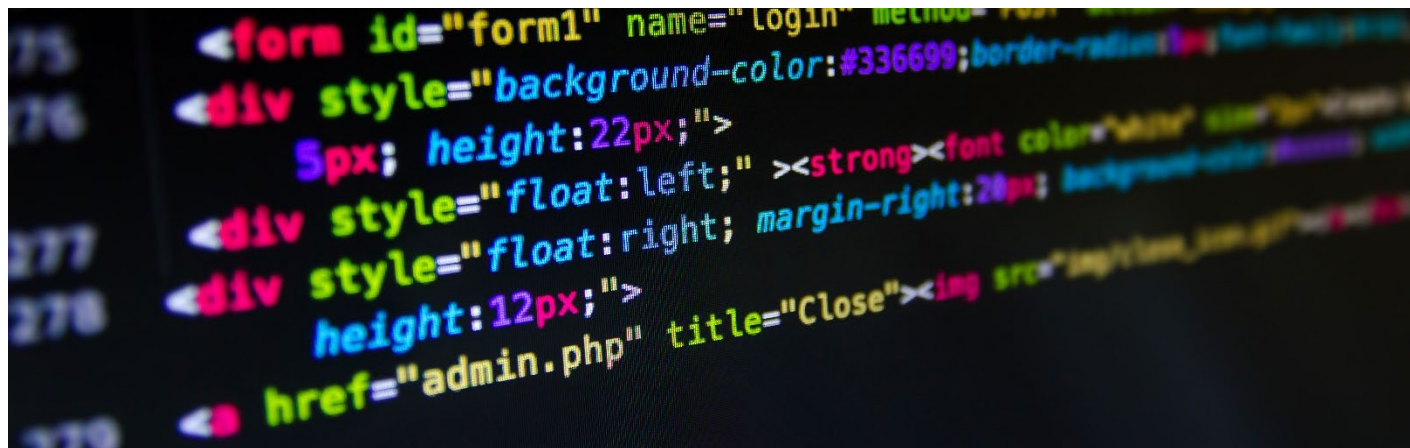
- Visual Studio Code - <https://code.visualstudio.com/>
- Notepad ++ - <https://notepad-plus-plus.org/downloads/>
- Sublime - <https://www.sublimetext.com/>

Cu plată:

- PhpStorm - <https://www.jetbrains.com/phpstorm/>
- WebStorm - <https://www.jetbrains.com/webstorm/>

**Observație:** În cadrul cursului vom folosi Visual Studio Code, însă nu este obligatoriu, fiecare cursant este încurajat să folosească editorul dorit.

**2. Prin folosirea editorului text pus la dispoziție de sistemul de operare.** Această metodă nu prea mai este folosită în ziua de azi, dar este bine de știut.

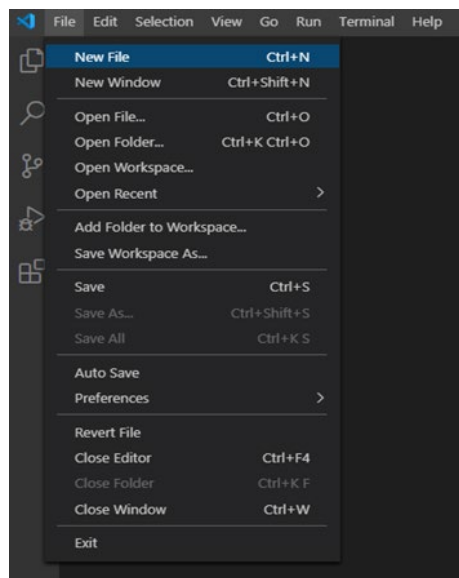


## 1.6 Cum creăm prima noastră pagină web?

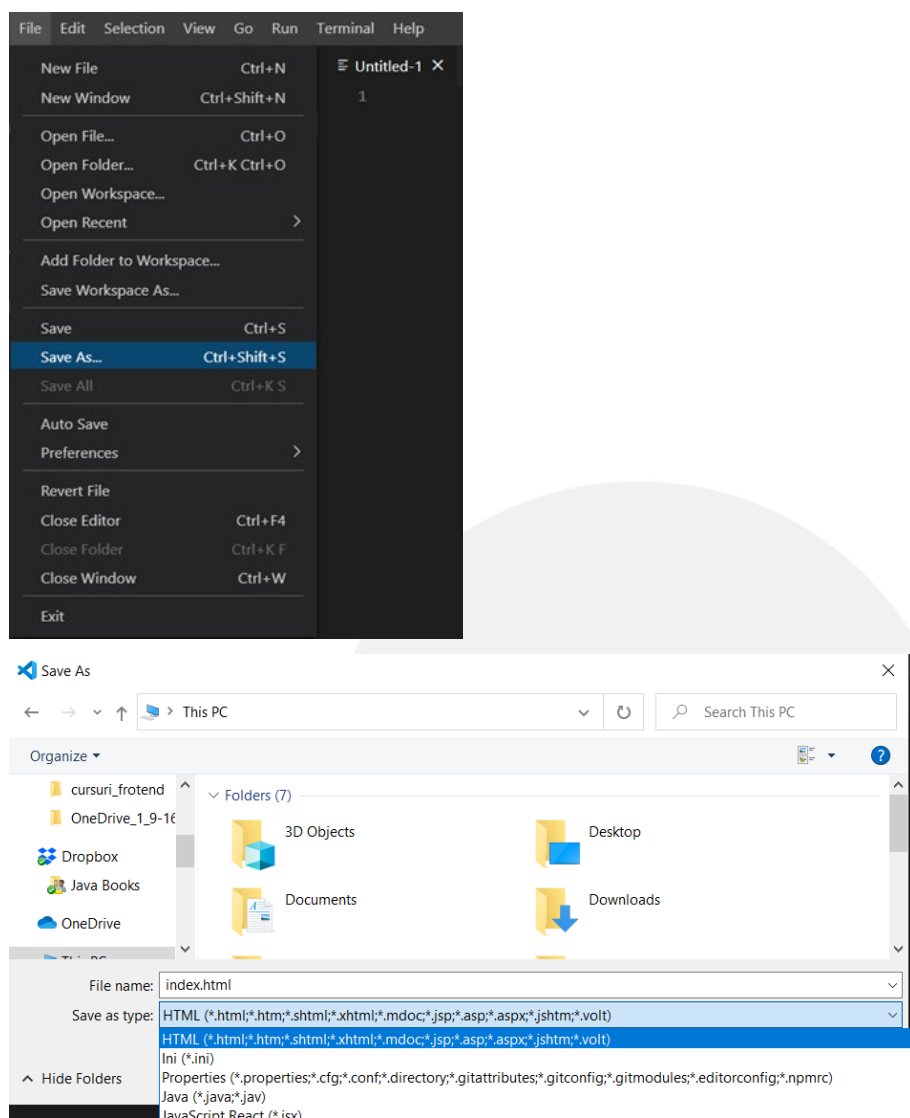
Deoarece este vorba de o pagină de tip client, nu avem nevoie să instalăm nimic, doar creăm un simplu fișier cu extensia .html, apoi deschidem acea pagină direct în browser conform următoarelor metode.

### 1. Pentru prima noastră metodă, folosim Visual Studio Code.

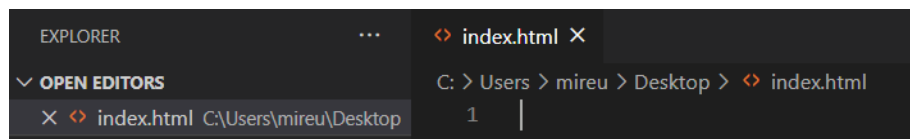
Deschide Visual Studio Code, mergem la secțiunea File și click pe New File. Ni se va deschide un fișier gol cu titlul "Untitled-1".



Mergem din nou la File alegem "Save As" și ni se va deschide următoarea fereastră.



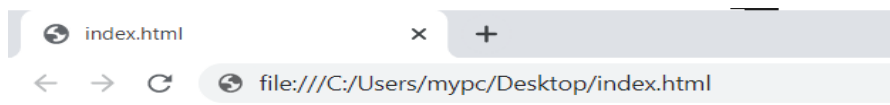
Alegem locația unde să fie salvat și un nume pentru fișierul nostru, cel mai des este folosit este "index" pentru pagina principală. Și la "Save as type" alegem extensia .html.



În urma salvării putem observa că extensia fișierului s-a schimbat, iconița < > ne indică faptul că este un fișier .html și putem observa și calea spre locul unde este salvat fișierul.

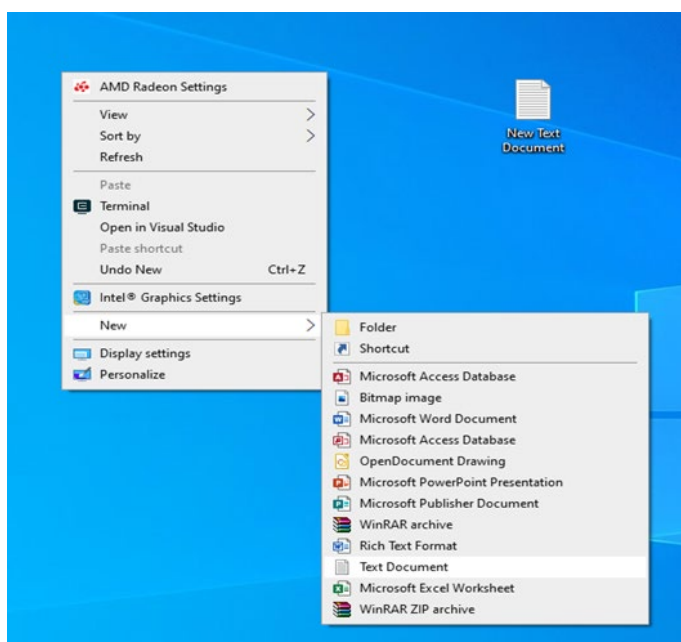


Putem observa că iconița fișierului de pe PC este și ea schimbată în cea a browserului default. În cazul prezentat este Chrome, în cazul vostru poate fi Firefox, Opera, Safari etc.



Dacă apăsăm dublu click pe fișier acesta se deschide în browser dar nimic nu este afișat din cauză că nu am adăugat nici o linie de html în fișierul creat. În cursul următor vom adăuga primele noastre linii de HTML.

2. Pentru cea de a doua metodă vom folosi **editorul text preinstalat** de sistemul de operare Windows: Notepad.



În momentul în care dam click pe Text Document, se va crea un fișier.txt, noi trebuie să avem doar fișier.html, este important deoarece altfel nu va funcționa.

Deschidem fișierul text, selectăm File, Save As și selectăm extensia .html.

## 1.7 Cum învățăm eficient?

Atunci când ne dorim cu adevărat, când suntem concentrați, ideal ar fi să putem învăța oriunde, însă este recomandat să învățăm în același loc și la ore fixe, alegând o încăpere liniștită, având o poziție cât mai corespunzătoare atunci când ne așezăm la o masă/birou, dar și o stare de spirit potrivită.

Este bine să înlăturăm orice ne poate distra, punând la o parte orice stă între noi și ceea ce urmează să citim, sau să scriem. Este necesar să înveți ca un obicei, într-un colț de cameră special amenajat pentru asta (un birou gol, într-o sufragerie). Faptul că îți schimbi locul, dintr-o cameră în alta, dintr-un mediu în altul, poate duce la scăderea randamentului de învățare. Este de asemenea recomandat să evităm pe cât mai mult posibil crearea haosului și să încercăm să avem un mediu curat și ordonat.

Chiar dacă consideri că nu mediul în care înveți îți afectează rezultatul învățării, este indicat să acorzi o atenție la spațiul pe care ți-l alegi și creezi pentru studiu. Sunete, mirosuri, un birou dezordonat, încărcat creează în mod inevitabil un disconfort. Ambientul trebuie doar să stimuleze într-un mod eficient. Orice distragere exterioară sau interioară cauzează oboseală și supraîncărcare.

Câteva elemente ce influențează pozitiv învățatul:

- ordine pe masa de lucru
- materiale folosite în timpul învățării (cărți, creioane, instrumente)
- așeza masa astfel încât să nu fie cu privire spre fereastră
- ia măsuri astfel încât să nu fii deranjat de orice zgomot
- camera să fie aerisită, ventilată, mereu curată
- temperatura optimă: 22-24 grade vara, iarna 18-20 grade
- umiditatea prea multă împiedică transpirația, iar aerul uscat irită
- nu înveța când ești obosit fizic sau mental
- înaintea fiecărei ședințe este bine să faci câteva exerciții de respirație
- este bine să fi orientat cu fața către răsărit
- iluminarea corectă, prea multă sau prea puțină lumină influențează învățatul, nu te poți concentra, este bine să îmbini lumina fluorescentă cu cea incandescentă (neon, bec), însă cea mai bună lumină este cea naturală, motiv pentru care învățatul este cel mai eficient ziua
- lumina trebuie să vină din partea stângă, puțin din spate, astfel încât să nu creeze reflexie pe foaie sau monitor
- trebuie evitate variațiile de lumină (să fie totul uniform)
- albul în general obosește, se recomandă o nuanță mai închisă, o tentă crem

Numeroase studii au dovedit o performanță mai ridicată a celor ce folosesc diferite culori față de cei ce folosesc doar negru pe o foaie albă ca și culoare, dovedindu-se astfel că un mediu cromat este mai eficient decât unul acromatic.

Principalele culori sunt: roșu, roșu aprins, portocaliu, galben, verde, albastru, violet, albul, maro, negrul.



- **Roșul** este o culoare foarte caldă, activează respirația, stimulează pofta de mâncare, stimulează diferite stări de neliniște în funcție de nuanța pe care o are.
- **Roșu Aprins** generează furie, nuanțele închise de roșu afectează dispoziția negativ, însă nuanțele ușoare oferă vigoare, energie, **roșu deschis** (roz), calmează, creând o ambianță odihnitoare și optimistă.
- **Portocaliul** este o culoare caldă, plăcută, care încurajează răbdarea, stimulează atenția, generează optimism și veselie, fiind considerat stimulator emotiv.
- **Galbenul** stimulează sistemul nervos, munca intelectuală, adoptând o stare de vigilență, concentrare sporită întărind memoria, îmbunătățind vederea. Generează căldură, intimitate și veselie.
- **Verdele** favorizează deconectarea, scade presiunea arterială, generează liniște, recomandat pentru combaterea emoțiilor negative.
- **Albastrul** favorizează dezvoltarea proceselor de inhibiție și încetinește ritmul activității, inspirând o stare a destinderii, scade presiunea sanguină și tonusul muscular, calmează respirația și pulsul.
- Lumina nocturnă de **albastru-deschis** are efect de relaxare, favorizând somnul.
- **Violetul**, culoare rece, neliniștită, generează sentimentele contradictorii (optimism nostalgic), **violetul deschis** este considerat plăcut, liniștit.
- **Albul** inspiră puritate, răceală, dă senzație de calm, liniște.
- **Maroul** este considerat sedativ, echilibrat (mai ales în combinație cu tonuri închise).
- **Negrul** generează și menține o stare negativă de neliniște, depresie.

## Temă

1. Să descărcați și să instalați un editor IDE prezentat mai sus.
2. Să creați un fișier HTML prin cele 2 metode menționate mai sus.

