**MINISTERUL EDUCAȚIEI AL REPUBLICII MOLDOVA**

**UNIVERSITATEA DE STAT “ALECU RUSSO” DIN BĂLȚI**

**FACULTATEA DE ȘTIINȚE REALE, ECONOMICE ȘI ALE MEDIULUI**

**CATEDRA DE MATEMATICĂ ȘI INFORMATICĂ**

**PROGRAMAREA WEB**

**REFERAT LA DISCIPLINA “INFORMATICA GENERALĂ”**

**Autor:**

Studentul grupei IT11Z

**Curbăt Anatolie**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Conducător științific:**

Olesea SKUTNIȚKI

magistru, lect. univ.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**BĂLȚI, 2022**

CUPRINS

[**INTRODUCERE** 3](#_Toc103596809)

[**1.** **PROGRAMAREA SITE-URILOR WEB** 4](#_Toc103596810)

[1.1. Conceptul de site web 4](#_Toc103596811)

[1.2. Tipuri de site-uri web 5](#_Toc103596812)

[**2.** **LIMBAJE DE PROGRAMARE UTILIZATE ÎN PROGRAMAREA ȘI PROIECTAREA SITE-URILOR WEB** 7](#_Toc103596813)

[2.1. Utilizarea limbajul HTML 7](#_Toc103596814)

[2.2. Utilizarea limbajul CSS 8](#_Toc103596815)

[2.3. Utilizarea limbajul JavaScript 9](#_Toc103596816)

[2.4. Utilizarea limbajul PHP 10](#_Toc103596817)

[**3.** **DEZVOLTATORI WEB** 12](#_Toc103596818)

[3.1. Conceptul și tipurile de dezvoltatori web 12](#_Toc103596819)

[3.2. Programatori Web de front end 12](#_Toc103596820)

[3.3. Programatori Web de back end 13](#_Toc103596821)

[3.4. Programatori Web Full stack 14](#_Toc103596822)

[**CONCLUZII** 15](#_Toc103596823)

[**Bibliografie** 16](#_Toc103596824)

# **INTRODUCERE**

De la mijlocul anilor 1990 , programarea web a fost una dintre industriile cu cea mai rapidă creștere din lume: în 1995 existau mai puțin de 1.000 de companii în acest sector numai în Statele Unite , dar până în 2005 erau deja peste 30.000, având o dezvoltare puternică în anii 2000 odată cu apariția Web 2.0 și răspândirea aplicațiilor Web.

Creșterea acestei industrii este determinată de perspectiva pieței de vânzare a produselor și serviciilor către clienții lor pentru a automatiza fluxul de lucru; sectorul programării web este, prin urmare, în general un sector înfloritor și în continuă dezvoltare.

Pe măsură ce cererea a crescut, a crescut și oferta, reducând prețurile. De asemenea, au fost dezvoltate platforme publice gratuite. Un exemplu este pachetul LAMP ( Linux , Apache , MySQL , PHP ). Dezvoltatorii web își propun să construiască aplicații care în mod tradițional erau disponibile numai în mediul local. Acest lucru a permis nașterea de noi metode de comunicare, descentralizarea informațiilor și distribuirea conținutului pe net.

Un exemplu de transformare pe care programarea web a adus-o în comunicații și comerț este comerțul electronic (de exemplu, eBay ). Un alt exemplu al modului în care dezvoltarea de software pentru web a adus transformări clare în modul de comunicare sunt blogurile .

**Termenul de programare web** indică activitățile și tehnicile de programare , cu limbajele de scriptare respective, care permit crearea și dezvoltarea de aplicații pentru Web. Programarea web este împărțită în trei tipuri:

a) proiectarea și dezvoltarea pentru crearea de site-uri web , CMS pentru bloguri , forumuri și rețele sociale , aplicații sau portaluri pentru comerțul electronic ;

b) scripturi pe partea de client și pe partea de server ;

c) configurarea serverelor web.

Teoretic, programarea web poate fi împărțită în programare web partea client și programare web partea server, în funcție de faptul dacă dezvoltați front-end-ul sau back-end- ul aplicației web. Adesea dezvoltatorul trebuie să aibă grijă de ambele părți. Aplicațiile web destul de simple pot fi, de asemenea, construite în întregime cu logică din partea clientului (de exemplu, în întregime în JavaScript ). Partea din stratul de date, adică crearea bazei de date cu tabele și întreținere aferente, este adesea opera unui administrator de baze de date (DBA) [1].

1. **PROGRAMAREA SITE-URILOR WEB**

1.1. Conceptul de site web

Noțiunea site web provine din expresia engleză web site și desemnează o grupă de pagini web multimedia (conținând texte, imagini fixe, animații ș.a.), accesibile în Internet în principiu orișicui, de obicei pe o temă anume, și care sunt conectate între ele prin așa-numite hiperlinkuri. Diversele situri web pot fi create de către o organizație, o persoană particulară, instituții publice etc. Inițial noțiunea apărea în limba română scrisă sub două forme : sit web și site (ca în limba engleză). Lingvistul George Pruteanu considera că varianta de preferat este sait.

De obicei un sit web este administrat (creat, întreținut și actualizat) de către un așa-numit webmaster, dar există și alte posibilități:

* situl web se actualizează automat și permanent pe baza unei baze de date;
* paginile sale se creează în mod dinamic și automat în funcție de acțiunea utilizatorului în cadrul unei aplicații web;
* situl web se creează și e administrat chiar de către utilizatorii săi.

Un site web este alcătuit de regulă din mai multe pagini web. O pagină web este un document creat cu ajutorul limbajului de marcare HTML și (opțional) limbaje de programare cum ar fi PHP, ASP ș.a. fiind accesibil vizitatorilor prin intermediul protocolului HTTP, care transferă informația de la server la browser. Pagina web se numește așa deoarece, afișată pe un monitor, ea se aseamănă cu o pagină de ziar: de obicei paginile web au o lățime care încape în întregime pe ecran. În schimb, pagina poate fi chiar mult mai înaltă (adâncă) decât înălțimea ecranului, ea putând fi totuși ușor afișată cu ajutorul funcțiilor normale ale mausului și browserului folosite, prin "tragere" în sus și în jos. De asemenea, un sit web poate fi vizualizat pe orice dispozitiv conectat la Internet capabil să afișeze informații prin intermediul protocolului HTTP (unele telefoane mobile, PDA-uri, etc.).

Un site web este scris in limbajul HTML si este accesat prin intermediul unei aplicatii software, cunoscuta si sub numele de browser web. Exemple de browser-e web: Internet Explorer, Mozilla Firefox, Google Chrome, Opera, Safari etc. Exista o multitudine de browser-e web, insa cele mai des utilizate (insumand 99% din utilizatorii de Internet) sunt primele trei mentionate anterior. Felul in care limbajul HTML este interpretat si afisat pe ecran poate diferi de la un browser web la altul. Pentru ca o pagina web sa fie afisata identic, indiferent de faptul ca este vizualizata de un utilizator care foloseste Internet Explorer sau alt browser web, se face asa-numita verificare cross-browser, prin care se aplica mici artificii ale limbajului HTML [2].

Structura unui site (fig.1.1) porneşte de la Homepage, care reprezintă prima pagină a site-ului, se ramifică în secţiuni principale şi subsecţiuni, şi se termină în paginile de conţinut [3].

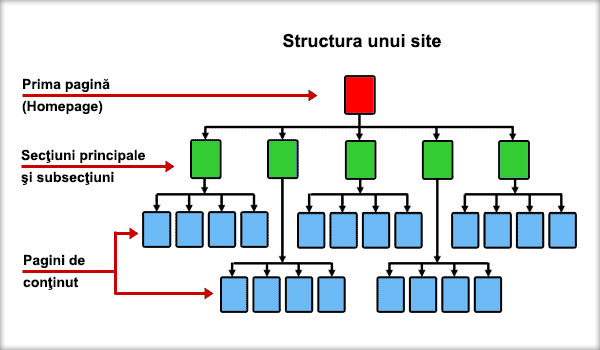


Fig.1.1. Structura unui site web.

1.2. Tipuri de site-uri web

Website-urile pot fi catalogate după foarte multe criterii: destinație , conținut , audiență , scop, tehnologie utilizată , număr de pagini etc. În funcție de modul în care este generat codul HTML al paginilor web ale unui un website, acesta poate fi static, dinamic sau combinat, adică poate conține **atât pagini web statice, cât și pagini web dinamice**.

**Un website static**, este format din pagini web statice, care sunt stocate pe serverul web în același format în care sunt transmise către browser. De obicei, acest format este HTML. Website-urile simple, de prezentare, sunt , de obicei, siteuri statice. Acest tip de website prezintă aceleași informații tuturor vizitatorilor. Siteul va prezența aceleași informații , structurate după formule consacrate, pentru perioade lungi de timp . Deși un website static poate fi actualizat periodic, acest proces se face manual și poate necesită anuminte cunoștințe specifice de limbaj HTML, de editare de imagine etc.

**Un website dinamic** este compus din pagini construite dinamic, al căror conținut este variabil, în funcție de diverse condiții din website sau condiții impuse de vizitator. Un website dinamic conferă mult mai multă flexibilitate comparativ cu unul static, dar presupune utilizarea unui limbaj de programare de nivel înalt precum PHP, ASP, Python, Perl etc. De cele mai multe ori, în combinație cu limbajul de programare, se utilizează și structuri de date avansate, denumite baze de date. Bazele de date pot fi de multe tipuri : statice, relaționale , orientate pe obiecte etc. Cel mai des utilizate sunt bazele de date relaționale , de tip SQL, și aici, că și implementare a acestui tip de baze de date, cel mă des utilizate sunt bazele de date mySQL. Că să ieșim puțin din partea tehnică a acestor tipuri de websiteuri, să menționăm că ele sunt, la ora actuală , cele mai răspândite deoarece permit automatizarea creării de pagini web, fiind mult mai simplu de a crea un șablon de pagină , un cod de generare a paginilor și o baza de date cu informațiile din pagini, decât să creezi sute de mii de pagini statice HTML individuale . Tipuri de website- uri dinamice: forumuri de discuții , bloguri, websiteuri de socializare, magazine online, reviste online etc [4].

1. **LIMBAJE DE PROGRAMARE UTILIZATE ÎN PROGRAMAREA ȘI PROIECTAREA SITE-URILOR WEB**

2.1. Utilizarea limbajul HTML

HyperText Markup Language (HTML) este un limbaj de marcare utilizat pentru crearea paginilor web ce pot fi afișate într-un browser (sau navigator). Scopul HTML este mai degrabă prezentarea informațiilor – paragrafe, fonturi, tabele ș.a.m.d. – decât descrierea semanticii documentului.(în figura 2.1 este reprezentatăstructura unui document HTML).

HTML este folosit pentru prezentarea unui conținut (text, imagine) într-o pagină web, furnizează mijloacele prin care conținutul unui document poate fi structurat și adnotat cu diverse tipuri de metadate și indicații de redare și afișare. Aceste indicațiile pot varia de la decorațiuni minore ale textului, cum ar fi culoarea sau sublinierea unui cuvânt ori introducerea unei imagini, până la adăugarea de elemente sofisticate, tabele, hărți de imagini, formulare și cod CSS sau scripturi JavaScript.

Metadatele pot include informații despre titlul și autorul documentului, informații structurale despre cum este împărțit documentul în diferite segmente, paragrafe, liste, titluri etc. și informații esențiale care permit ca documentul să poată fi legat de alte documente pentru a formă astfel hiperlink-uri.

Paginile HTML sunt formate din etichete sau tag-uri și au extensia „.html” sau „.htm”. În marea lor majoritate aceste etichete sunt pereche, una de deschidere <eticheta> și alta de închidere </eticheta>, mai există și cazuri în care nu se închid, atunci se folosește <eticheta />. Navigatorul web interpretează aceste etichete afișând rezultatul pe ecran. HTML-ul este un limbaj care nu face deosebire între litere majuscule și minuscule.

Pagina principala a unui domeniu este fisierul „index.html” respectiv „index.htm”. Această pagină este setată a fi afișată automat la vizitarea unui domeniu.De exemplu la vizitarea domeniului www.nume.md este afișată pagina www.nume.md/index.html [5].

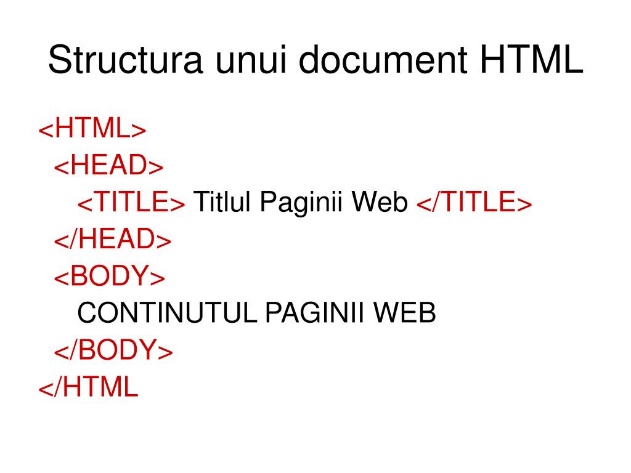


Fig.2.1. Structura unui document HTML.

2.2. Utilizarea limbajul CSS

Despre CSS se poate spune că este un limbaj de programare, care se axează pe stilul utilizat în prezentarea unui document scris într-un limbaj de marcare, cum ar fi HTML. Denumirea CSS provine din expresia Cascading Style Sheets și este o tehnologie de baza a World Wide Web, alături de HTML și JavaScript.

El este conceput pentru a permite stilizarea conținutului unei pagini web și a prezentării acesteia, inclusiv a aspectului, fontului și culorilor. Această procedura îmbunătățește în mod real accesibilitatea conținutului, oferă mai multă flexibilitate și control în ceea ce privește specificațiile caracteristicilor prezentării, permite că mai multe pagini web să partajeze formatarea și reduce repetarea și complexitatea conținutului structurat.De asemenea, formatează pagină în așa fel încât să fie fezabilă pe diferite tipuri de ecrane, prin voce, dar și pe dispozitivele tactile Braille. Mai mult, are reguli pentru formatarea alternativă în cazul în care conținutul este accesat de pe un dispozitiv mobil și modifică aspectul paginii în așa fel încât să poate fi vizibil din orice perspectiva.

CSS este unul dintre cele mai utilizate limbaje de stilizare de pe web, din mai multe motive:

* ajută la economisirea timpului – codul se poate scrie o singura data și apoi se poate reutiliza în mai multe pagini HTML. De asemenea, puteți defini un stil pentru fiecare element HTML și îl poți aplica la oricâte pagini web ai nevoie.
* paginile se încarca mai rapid – cu cât adaugi mai mult cod unei pagini, cu atât aceasta se vă încarca mai greu. CSS iți permite să utilizezi cât mai puțin cod printr-o anumită regula, ce poate fi aplicata la toate etichetele dintr-un document HTML. Astfel, daca ai una sau mai multe pagini de produse care ar trebui să aibă aceeași formatare, CSS te vă ajuta că aceasta procedura să fie realizata doar prin adăugarea unui cod.
* oferă o experiență mai buna a utilizatorului – pe lângă faptul că facilitează vizibilitatea unei pagini web din perspectiva utilizatorului, permite și formatarea foarte ușoară a acesteia. Atunci când ai butoane și text bine organizat, amplasate în locuri logice pentru cititor, experiență lui se îmbunătățește foarte mult.
* modificări ușoare de formatare – oricând ai nevoie să modifici anumite formate ale paginilor tale web, CSS iți facilitează acest lucru, fară a fi nevoie să remediezi fiecare pagina în parte. Trebuie să editezi doar foile de stil corespunzătoare și codul vă fi aplicat tuturor paginilor care le utilizează.
* este compatibil pe toate dispozitivele – nimic nu este mai neplăcut decât să nu poți accesa o pagina anume atunci când utilizezi un ecran mai mic. Toate paginile ar trebui să fie vizibile, smart, și foarte ușor de navigat, indiferent de dispozitivul care le accesează.
* standarde web globale – HTML are aproximativ aceleași atribute că și CSS, însă este considerat un limbaj învechit pentru stilizare. De aceea este indicat că în formatarea site-ului tău să folosești CSS, pentru că pe viitor paginile să fie compatibile cu toate browserele și să nu întâmpini probleme în acest sens.
* este superior HTML-ului – daca iți dorești să beneficiezi de o gama cât mai larga de atribute, CSS este limbajul de stilizare pe care trebuie să îl folosești. Te vă ajuta să oferi un aspect mult mai bine structurat și mai ușor de parcurs pentru utilizatori decât HTML. În plus, fata de HTML și alte limbaje accepta utilizarea CSS, inclusiv XHTML, XML simplu, SVG și XUL [6].

2.3. Utilizarea limbajul JavaScript

JavaScript a fost dezvoltat prima dată de către firma Netscape, cu numele de Live Script, un limbaj de script care extindea capacitățile HTML, oferă o alternativă parțială la utilizarea unui număr mare de scripturi CGI pentru prelucrarea informaţiilor din formulare și care adaugă dinamism în paginile web. După lansarea limbajului Java, Netscape a început să lucreze cu firma Sun, cu scopul de a crea un limbaj de script cu o sintaxă și semantică asemănătoare cu a limbajului Java, și din motive de marketing numele noului limbaj de script a fost schimbat în "JavaScript".

JavaScript a apărut din nevoia că logica și inteligența să fie și pe partea de client, nu doar pe partea de server. Dacă toată logica este pe partea de server, întreaga prelucrare este facută la server, chiar și pentru lucruri simple, așa cum este validarea datelor. Astfel, JavaScript îl înzestrează pe client și face că relația să fie un adevărat sistem client-server.

În general se consideră că există zece aspecte fundamentale ale limbajului JavaScript pe care orice programator în acest limbaj ar trebui să le cunoască :

1. **JavaScript poate fi introdus în HTML** - De obicei codul JavaScript este gazduit în documentele HTML și executat în interiorul lor. Majoritatea obiectelor JavaScript au etichete HTML pe care le reprezintă, astfel încât programul este inclus pe partea de client a limbajului. JavaScript foloseste HTML pentru a intra în cadrul de lucru al aplicațiilor pentru web.
2. **JavaScript este dependent de mediu** - JavaScript este un limbaj de scriptare; software-ul care ruleaza de fapt programul este browser-ul web (Firefox, Opera, Netscape Navigator, Internet Explorer, Safari, etc.) Este important să luăm în considerare această dependență de browser atunci când utilizăm aplicații JavaScript.
3. **JavaScript este un limbaj în totalitate interpretat** - codul scriptului va fi interpretat de browser înainte de a fi executat. JavaScript nu necesită compilări sau preprocesări, ci rămane parte integranta a documentului HTML. Dezavantajul acestui limbaj este că rularea durează ceva mai mult deoarece comenzile JavaScript vor fi citite de navigatorul Web și procesate atunci când user-ul apelează la acele funcții (prin completare de formulare, apăsare de butoane, etc). Avantajul principal este faptul că putem mult mai ușor să actualizăm codul sursă.
4. **JavaScript este un limbaj flexibil** - în aceasta privință limbajul diferă radical de C++ sau Java. În JavaScript putem declara o variabilă de un anumit tip, sau putem lucra cu o variabilă deși nu-i cunoaștem tipul specificat înainte de rulare.
5. **JavaScript este bazat pe obiecte** - JavaScript nu este un limbaj de programare orientat obiect, ca Java, ci mai corect, este "bazat pe obiecte"; modelul de obiect JavaScript este bazat pe instanță și nu pe moștenire.
6. **JavaScript este condus de evenimente** - mare parte a codului JavaScript răspunde la evenimente generate de utilizator sau de sistem. Obiectele HTML, cum ar fi butoanele, sunt îmbunătățite pentru a accepta handlere de evenimente.
7. **JavaScript nu este Java** - Cele doua limbaje au fost create de companii diferite, motivul denumirii asemănătoare este legat doar de marketing.
8. **JavaScript este multifuncțional** - limbajul poate fi folosit într-o multitudine de contexte pentru a rezolva diferite probleme: grafice, matematice, și altele.
9. **JavaScript evoluează** - limbajul evoluează, fapt pozitiv care însă poate genera și probleme, programatorii trebuind să verifice permanent ce versiune să folosească pentru ca aplicațiile să poată fi disponibile unui numar cât mai mare de utilizatori de browsere diferite.
10. **JavaScript acoperă contexte diverse** - programarea cu acest limbaj este îndreptată mai ales către partea de client, dar putem folosi JavaScript și pentru partea de Server. JavaScript este limbajul nativ pentru unele instrumente de dezvoltare web, ca Borland IntraBuilder sau Macromedia Dreamweaver [7].

2.4. Utilizarea limbajul PHP

PHP este un limbaj de programare. Numele PHP provine din limba engleză și este un acronim recursiv : Php: Hypertext Preprocessor. Folosit inițial pentru a produce pagini web dinamice, este folosit pe scară largă în dezvoltarea paginilor și aplicațiilor web.

PHP-ul a fost dezvoltat la sfârșitul anului 1994 de către Rasmus Lerdorf, și a fost utilizat în scopuri personale. Prima versiune care a fost folosita și de alti programatori a apărut la începutul anului 1995, fiind cunoscuta că Personal Home Page Tools. Aceasta versiune permitea înțelegerea câtorva funcții și macro-uri folosite în mod regulat pentru realizarea de guestbook-uri și de contoare. Acesta a fost momentul în care programatorii din țoață lumea au început să-și aducă contribuția la dezvoltarea limbajului. La sfârșitul anului 1997, PHP/FI era folosit în peste 50.000 de site-uri Web din întreagă lume. În 1997, Zeev Suraski și Andi Gutmans au rescris de la zero serverul de PHP/FI, iar noua versiune lansata pe piață a devenit cunoscuta sub denumirea de PHP3.

La mijlocul anului 1999, numărul de servere pe care rula PHP-ul ajunsese la 1.000.000. Se folosește în principal înglobat în codul HTML, dar începând de la versiunea 4.3.0 se poate folosi și în mod „linie de comandă” (CLI), permițând crearea de aplicații independente.

Este unul din cele mai importante limbaje de programare web open-source și server-side, existând versiuni disponibile pentru majoritatea web serverelor și pentru toate sistemele de operare. Conform statisticilor este instalat pe 20 de milioane de site-uri web și pe 1 milion de servere web. Este disponibil sub Licenṭa PHP ṣi Free Software Foundation îl consideră a fi un software liber [8].

1. **DEZVOLTATORI WEB**

3.1. Conceptul și tipurile de dezvoltatori web

În domeniul informaticii, un dezvoltator web (în engleză web developer), este o figură profesională care se ocupă cu dezvoltarea aplicațiilor web, în special dezvoltarea codului sursă de programare, optimizarea bazelor de date și a limbajelor de marcare prin intermediul tehnicilor de programare web.Acești programatori pot fi programatori de front end, de back end sau full stack.

Un dezvoltator web lucrează în echipe cu figuri precum web designeri, webmasteri și manageri de conținut web cu care nu ar trebui să fie confundat. Un dezvoltator web des, nu proiectează grafica unui site, nu se ocupă de conținut sau de crearea de fișiere multimedia, cum ar fi video sau audio, dar se ocupă în principal de asamblarea, structurarea și automatizarea unui site. Acest lucru se datorează faptului că, în general, dezvoltatorul de web se poate baza pe cunoștințe mult mai tehnice decât celelalte figuri din industrie. Un dezvoltator de web, de fapt, trebuie să dețină cunoștințe specializate legate de protocoalele de rețea, servere web și formele lor principale, baze de date, securitatea calculatoarelor și limbaje de programare, precum și cunoștințe avansate a celor mai comune limbaje de markup pe web. Foarte des dezvoltatorii web sunt, de asemenea, designeri web, sau au fost în trecut.

Cu dezvoltarea web se poate înțelege dezvoltarea unei pagini web simplă, statică, sau dezvoltarea celor mai complexe aplicații web din web-ul dinamic, cum ar fi comerțul electronic, dar și crearea de rețele sociale. Un bun dezvoltator de web poate realiza orice aplicație web, exploatând cele mai relevante tehnologii atât pe partea de server, cât și pe partea clientului.

Figura dezvoltatorului web devine din ce în ce mai importantă din cauza tendinței actuale de a transfera aplicații desktop pe web și a permite utilizatorului să acceseze datele lor independent de computer și de locul în care lucrează. În ultimii ani, a devenit din ce în ce mai clar că un site web poate avea succes atunci când este util pentru utilizator, atunci când este capabil să efectueze funcții avansate și personalizate. Fără suportul unui dezvoltator web, aceste rezultate sunt greu de obținut.

3.2. Programatori Web de front end

Un Programator Web de front end este persoana care implementează designurile web prin limbaje specifice precum HTML, CSS și JavaScript. Spre deosebire de programatorii de front end, programatorii web de back end se ocupă de programarea din spate, precum bazele de date. Dacă intri pe orice site, poți vedea oriunde rezultatul muncii unui Programator Web de front end - în navigarea paginii, aranjarea paginii, inclusiv modul în care site-ul respectiv arată pe telefonul mobil.

Practic, un Programator Web de front end este responsabil elementele de conținut ale site-ului să fie clare, vizibile și la locul potrivit. În unele cazuri, aceștia au și abilități de content writing, creând în același timp și conținutul site-ului.

Programatorii web de front end se asigură că toate culorile site-ului sunt potrivite, mai ales cele care conțin text, culorile de fundal și titlurile. Cei mai buni programatori web de front end au și cunoștințe de design, putând astfel să modifice diverse aspecte pe parcurs.

Nu în ultimul rând, se asigură că toate linkurile sunt formatate corect, că toate butoanele funcționează corect și că site-ul este funcțional și că arată bine. De multe ori, tot ei sunt responsabili și de designul versiunii de mobil a site-ului, dar și de funcționarea acestuia pe toate tipurile de browser.

3.3. Programatori Web de back end

Deși poate părea că programatorii web de front end au o viață destul de dificilă în a se asigura că un site arată bine, funcționează bine și că afișează conținutul corect, programatorii web de back end au însă o viață mult mai grea. În timp ce programatorii web de front end sunt responsabili de programarea cu care interacționează clienții, programatorii de back end se ocupă de partea de servere.

Acest lucru înseamnă că ei trebuie să creeze codul și programele care operează serverele, bazele de date și alte aplicații ale acestora. În calitate de Programator Web de back end, cea mai mare calitate este abilitatea de a crea un program curat și eficient care face tot ce îți dorești în cel mai scurt timp. Din moment ce viteza site-ului este foarte importantă atunci când se ajunge la partea de SEO, aceasta este foarte importantă în back end development.

Pentru a explica ce face un Web Developer, este important de știut că programatorii de back end folosesc o gamă variată de limbaje asociate cu serverele pentru a construi programe complicate. Printre cele mai populare limbaje folosite în back end development se numără PHP, Python, Java și Ruby. JavaScript începe să devină și acesta popular printre programatorii back end, iar SQL este și el des folosit pentru a manageria și a analiza informațiile din cadrul bazelor de date.

Din moment ce fiecare website are alte nevoi, un Programator Web de back end trebuie să fie flexibil și capabil să creeze diverse programe. De asemenea, aceștia trebuie neapărat să aibă o înțelegere clară și aprofundată a limbajelor pe care le folosesc. Acest lucru este foarte important pentru a se asigura că vin soluția cea mai eficientă pentru crearea programului respectiv, asigurându-se de asemenea că este sigur și ușor de întreținut [9].

3.4. Programatori Web Full stack

Un Full stack developer este cel mai în măsură să conducă un proiect de web development de la început la bun sfârșit, acesta având experiență atât pe partea de strategie, cât și pe practicile din domeniu.

Având cunoștințe atât de front end, cât și de back end, pot vedea mai bine „imaginea de ansamblu” și sunt de cele mai multe ori cea mai bună alegere atunci când vine vorba de proiecte medii în termeni de complexitate sau care nu au un buget foarte mare.

Conceptul de full stack se referă la cunoașterea și utilizarea mai multor tehnologii și baze de date atât pentru dezvoltarea front-end (client software), cât și pentru dezvoltarea back-end (server software).

De aceea, este un rol foarte căutat de angajatori, fiind considerat echivalentul unei echipe de front-end și back-end, capabil să analizeze și să dezvolte optim proiectul în ansamblu.

Rolul de full stack developer presupune să știi cel puțin 5-6 limbaje de programare sau frameworks (față de 4, în medie, cerute pentru un programator specializat), fără a fi limitat la acestea. Nu este nevoie să fii expert în toate, dar e important să poți lucra cu toate.

Avantaje și dezavantaje:

Există multe avantaje în cariera de full stack developer. Datorită cunoștințelor variate și a tehnologiilor cunoscute, poți avea grijă de un proiect de la început până la final. Asta îți oferă mai multă autonomie, putere de decizie, rapiditate în workflow și, desigur, un salariu mai bun.

În plus, poți să-ți ajuți colegii indiferent de rolurile lor în proiect pentru că te poți adapta la cerințele de front-end ori back-end, după caz. Ești un bun comunicator și ai abilități de management, prin urmare poți reduce timpul petrecut pe proiect și pe comunicare, cât și costurile în sine, ceea ce te face un membru valoros al oricărei echipe.

Ca dezavantaje, ar putea fi faptul că trebuie să fii atent la toate aspectele proiectului, cât și foarte disciplinat. De asemenea, soluția pe care o propui și o implementezi poate să nu fie cea optimă [10].

# **CONCLUZII**

Dezvoltarea web este un proces de creare a unui site web pentru o afacere sau o organizație. În lumea digitalizării, în care majoritatea sarcinilor noastre zilnice nu pot fi îndeplinite fără ajutorul tehnologiilor web, crearea unui site web pentru o afacere este la fel de importantă ca și crearea unei strategii pentru afacere. Prezența digitală a unui brand nu doar promovează afacerea, ci oferă și o oportunitate de a se extinde pe tot globul.

Avantaje ale dezvoltării web:

* Impresie de lungă durată;
* Economisire de timp și bani;
* Extinderea razei de acțiune;
* Mai multe vânzări;
* Feedback valoros.

Dezvoltarea web este un instrument puternic pentru afaceri. Un site web bine conceput și receptiv are mari șanse să câștige încrederea utilizatorilor și să atragă o audiență mai mare. Atunci când un utilizator capătă încredere într-o afacere și ajunge să cunoască marca și valorile acesteia, lasă o impresie despre marcă.

Dezvoltarea web este ca o investiție unică. Nu trebuie să cheltuiți bani în plus pentru a obține vânzări sau pentru a promova marca. Dezvoltarea web economisește timp și bani, deoarece puteți ajunge la mii de oameni și totuși nu trebuie să cheltuiți bani pe bannere sau alte activități promoționale.

Dezvoltarea web oferă o oportunitate de a ieși dintr-o anumită zonă și de a extinde afacerea pe tot globul. Lumea digitală nu are nicio restricție, ceea ce permite afacerii să atragă mai mulți utilizatori din întreaga lume și să crească vânzările pentru marcă.

Majoritatea afacerilor au profitat de oportunitatea de a profita de vânzările online și de a obține un avantaj suplimentar față de concurenți. Potrivit unui studiu, 64% dintre consumatori preferă să facă cumpărături online. Prezența digitală nu numai că oferă oportunitatea de a se extinde, dar și de a atrage mai mulți utilizatori și lead-uri pentru o afacere, ceea ce aduce automat mai multe vânzări pentru o afacere.

Dezvoltarea web oferă o platformă de interacțiune între afacere și client. Clientul poate afla informații despre afacere și, de asemenea, poate împărtăși feedback-ul despre produs și servicii. În mod similar, afacerea poate ști unde poate lucra și ce le place consumatorilor în legătură cu marca.

Dezvoltarea web este un instrument puternic pentru afaceri, deoarece oferă diverse oportunități pentru a ridica nivelul afacerii. Aceasta a deschis ușile pentru ca lumea să interacționeze și să extindă afacerea pe tot globul fără nicio restricție.

# **Bibliografie**

1. Programare Web [online] [accesat 26 aprilie 2022]. Disponibil: <https://koaha.org/wiki/Programmazione_Web>
2. Site Web [online] [accesat 26 aprilie 2022]. Disponibil: <https://ro.wikipedia.org/wiki/Site_web>
3. Structura unui site web [online] [accesat 26 aprilie 2022]. Disponibil: <https://www.whd.ro/ce-este-un-site-web>
4. Tipuri de site-uri web [online] [accesat 26 aprilie 2022]. Disponibil: <https://www.websiteuri.ro/ce-este-un-website.html>
5. Limbajul HTML [online] [accesat 29 aprilie 2022]. Disponibil: <https://github.com/Tolea2001/Referat-IG/commits/main>
6. Limbajul CSS [online] [accesat 29 aprilie 2022]. Disponibil: <https://www.thc.ro/blog/ce-este-css-si-pentru-ce-se-poate-utiliza/>
7. Limbajul JavaScript [online] [accesat 29 aprilie 2022]. Disponibil: <https://web.ceiti.md/lesson.php?id=16>
8. Limbajul PHP [online] [accesat 29 aprilie 2022]. Disponibil: <https://ro.wikipedia.org/wiki/PHP>
9. Dezvoltatori Web [online] [accesat 29 aprilie 2022]. Disponibil: <https://ro.wikipedia.org/wiki/Dezvoltator_web>
10. Dezvoltatori Web [online] [accesat 29 aprilie 2022]. Disponibil: <https://it.emoldova.org/community/technology/ce-este-programarea-web/>