

TEHNOLOGII WEB

Natalia Pleșca, lect. univ

REPARTIZAREA ORELOR

Ciclul (L-licența, M-masterat)	L	Codul cursului	S.03.O.020.63	Anul	II	Semestrul	III
Nr. credite	4	Limba de instruire	Rom/ Rus	Forma de evaluare finală (E – examen, V- verificare)			E
Ore de contact direct		60	Ore studiu individual	60	Total ore pe semestru		120

Numărul total de ore (pe semestru) de contact direct (<i>C-curs teoretic, S-seminar, Lab-activități de laborator</i>)			
Total	C	S	Lab.
60	14	-	46

TEME PROPUSE PENTRU STUDIU

- Internet-ul - mediu uzual pentru comunicare
- HTML. Tag-uri HTML
- Implementarea tag-urilor HTML în documente HTML
- Stiluri CSS. Implementarea stilurilor în documente HTML
- CSS3 și HTML5
- JavaScript. Implementarea JavaScript-urilor în documente HTML
- Framework-ul front-end Bootstrap

MATERIALELE CURSULUI

- Toate materialele cursului – lecțiile teoretice și cerințele pentru lucrările de laborator - sunt disponibile pe platforma Moodle
- Tot acolo găsiți programul cursului, cerințele pentru elaborarea lucrului individual

PS: Adresa mea de mail natalia-plesca@yandex.ru

PRODUS PROPUS PENTRU “LUCRUL INDIVIDUAL”

Proiectul de investigație

Artefactele ce trebuie prezentate la sfârșitul semestrului sunt: un site creat de dvoastră și raportul (darea de seamă) despre elaborarea site-ului

Se va alege un domeniu, organizație etc. pentru care se va elabora un site static (sau tehnologia DHTML), folosind HTML+CSS+JavaScript

Raportul elaborării site-ului și site-ul se va prezenta profesorului, la orele de laborator (conform cerințelor înaintate), cel târziu înaintea terminării semestrului

CALCULAREA NOTEI MEDII

■ Nota semestrială = $\frac{LI+T1+T2+RC}{4}$, unde

□ LI = Site + Raport

□ T1 – testare 1

□ T2 – testare 2

□ RC – reușita curentă (activitatea la orele de laborator)

REFERINȚE BIBLIOGRAFICE

BIBLIOGRAFIE RECOMANDATĂ

- Вейл Э., HTML5, Разработка приложений для мобильных устройств. — СПб.: Питер, 2015 — 480 с.: ил. — (Серия «Бестселлеры O'Reilly»). ISBN 978-5-496-01125-9

Referințe în Internet:

- <http://www.w3schools.com> (Cursuri: HTML, CSS, JavaScript. Bootstrap)
- <https://www.javascript.com/>
- <https://www.codecademy.com/courses/web-beginner-en-HZA3b/o/1>
- <https://w3c.github.io/html/>
- <http://www.intuit.ru/studies/courses/3734/976/lecture/27460>

TEMA 1: INTRODUCERE. LIMBAJUL DE MARCARE HTML

Conținut lecție

▪ Introducere

- serviciul web
- tehnologii web
- browsere web
- modelul client-server

▪ Limbajul de marcare HTML

- Documente web
- Structura de bază a unui document web
- Editoare de cod

3 SINONIME... ASOCIERI

Pentru noțiunile

- Internet
- Web
- Browser Web
- Server web
- Site web **VS** pagină web
- FrontEnd

INTERNETUL

INTERNET - **inter-networks** (inter-rețele)

▪ **Internetul**

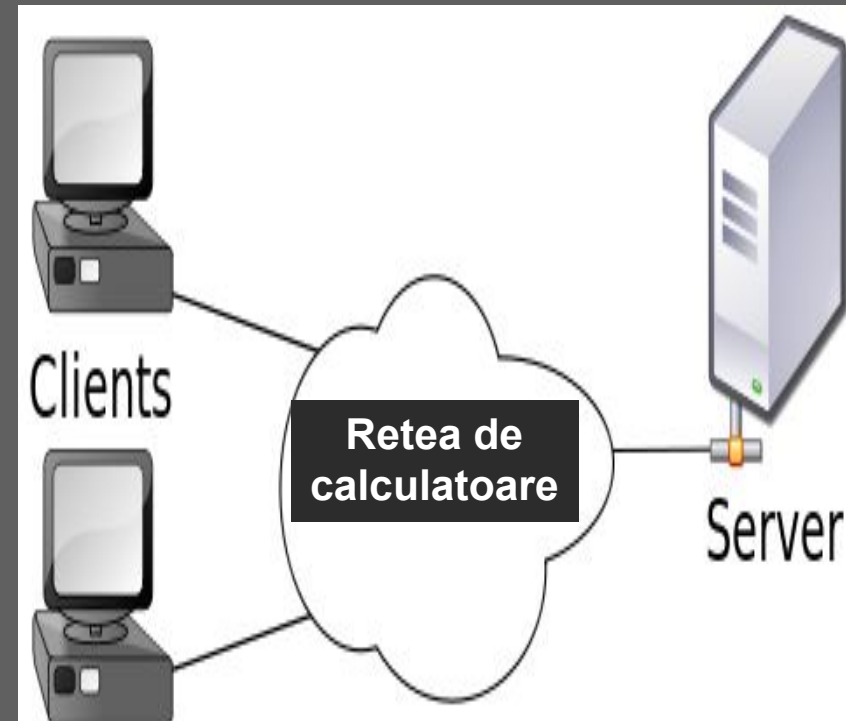
- rețea de rețele de calculatoare
- rețea compusă din rețele mai mici sau PC-uri individuale interconectate, ce acoperă tot globul
- rețea formată dintr-un număr mare de servere de Internet, legate între ele, la fiecare server fiind conectate un număr de stații sau alte rețele

▪ **Internet of Things (IoT) - ?**

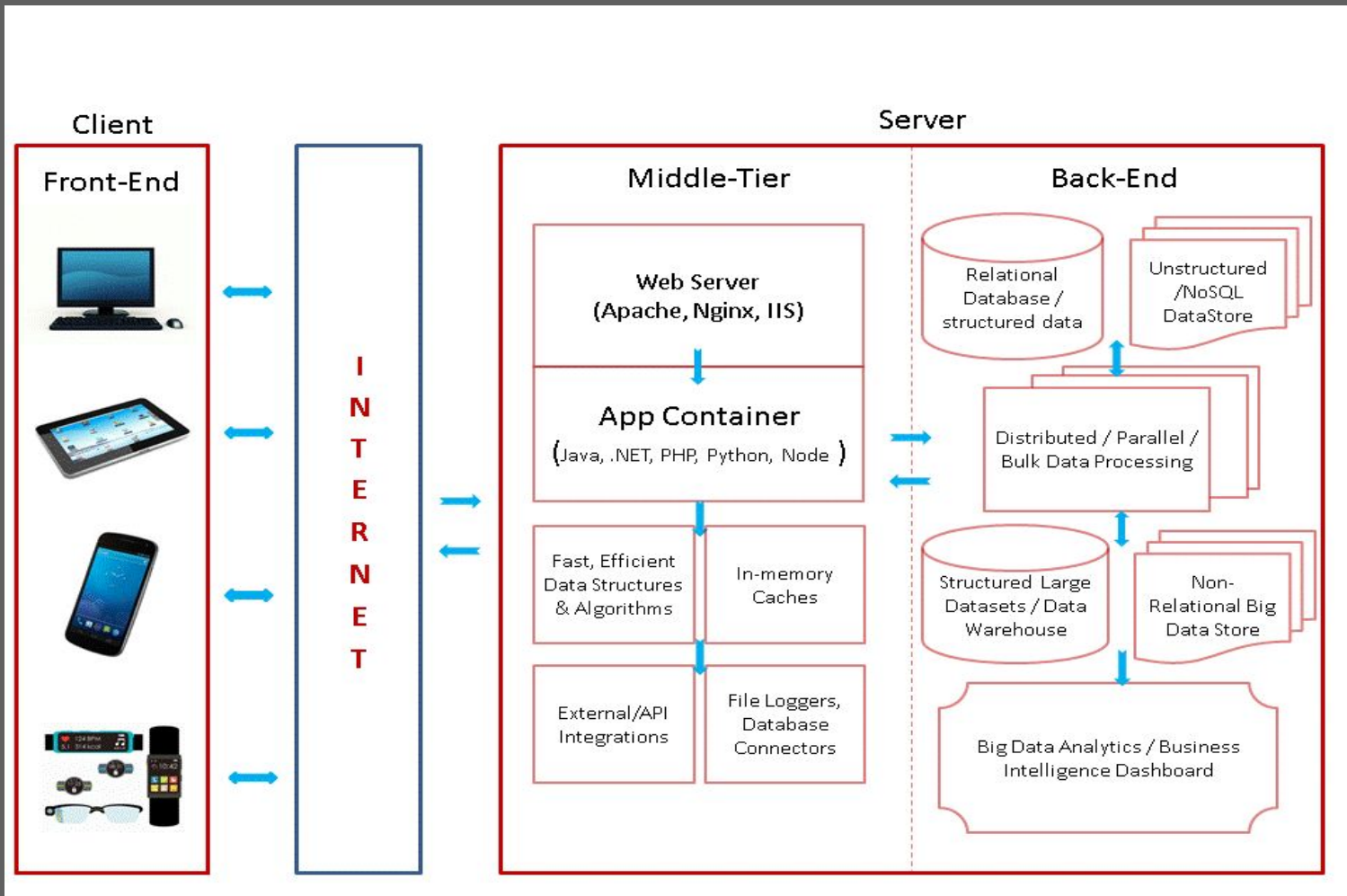


MODELUL CLIENT-SERVER

- Interacțiunea dintre calculatoarele și programele care oferă servicii Internet se bazează pe **modelul client-server**
 - Pe PC-ul beneficiarului de serviciu Internet rulează aplicația-client (apelează la resursele serverului), iar pe PC-ul furnizorului de servicii rulează aplicația-server (partajează resursele cu clienții)



INTERACȚIUNEA CLIENT-SERVER



WEB – UNUL DINTRE SERVICIILE DE BAZĂ ÎN INTERNET

Serviciul **WWW** (**World Wide Web**) sau simplu Web – partea multimedia a rețelei globale

- se utilizează pentru prezentarea și căutarea informațiilor în Internet
- informațiile sunt prezentate utilizatorului prin intermediul paginilor Web
- Web-ul constă din milioane de pagini care se pot afișa pe ecranul utilizatorului. Aceste pagini, numite *pagini Web*, pot conține texte, imagini grafice, animații, fișiere audio și video precum și hyperlink-uri
- Responsabil de web este consorțiul WWW: <https://www.w3.org/>
- Recomandările W3C sunt orientate spre standardizarea tehnologiilor web

EXEMPLE DE TEHNOLOGII FOLOSITE PENTRU A SCRIE PARTEA CLIENT A SITE-ULUI/ APLICAȚIEI WEB

Scripturile-client în general sunt vizibile de orice vizitator al unui site sau utilizator a unei aplicații web (din meniul de context, când acesta selectează "Vizualizați sursa"). Tehnologii uzuale folosite pentru frontend:

- HTML
- CSS
- JavaScript
- Ajax (JavaScript asincron și XML)
- jQuery (JavaScript Framework Library - utilizat în mod obișnuit în dezvoltarea Ajax)
- Angular, React, Vue (JavaScript Frameworks) etc.

EXEMPLE DE TEHNOLOGII FOLOSITE PE PARTEA SERVER

Scripturile server-side nu sunt vizibile sau accesibile vizitatorilor (utilizatorilor).
Câteva exemple de tehnologii utilizate în Server-Side scripting:

- PHP (limbaj foarte popular pentru server-scripting - Open Source - redistribuire gratuită, de obicei se combină cu baze de date MySQL)
- Framework-uri PHP: Symfony, Laravel, Zend etc.
- ASP (tehnologie a companiei Microsoft, Web Server IIS)
- ASP.NET (Microsoft Application Framework - succesor al ASP)
- Ruby on Rails (program de aplicații web al Ruby - redistribuire gratuită)
- Perl (limbaj de programare de nivel înalt și redistribuire gratuită a limbajului de scripting server-side - și-a pierdut popularitatea în favoarea PHP)
- Framework-uri bazate pe limbajul Java: Spring, Hibernate etc.
- Frameworkuri și platforme bazate pe JavaScript: Express.js (bazat pe node.js), Node.js etc.
- Python (limbaj de programare la nivel înalt), frameworkuri Python, GO etc.

BROWSERE WEB

- Oferă accesul la WWW și la celelalte servicii Internet, utilizând modelul client-server
- Permite „răsfoirea” paginilor web, aflate pe diverse servere
- Utilizatorul folosește un program de tip client, care se numește program pentru răsfoire sau explorare (în engleză **browser**), în română **navigator**
- Browser-ul - o aplicație software ce permite utilizatorilor să afișeze text, grafică, video, muzică și alte informații plasate pe o pagină Web, dar și să comunice cu furnizorul de informații
- Browserele au rolul de a ”traduce” - interpreta din limbaj HTML (CSS, JavaScript) într-un limbaj cunoscut utilizatorilor, permițând vizualizarea informațiilor

ACȚIUNI POSIBILE ÎN NAVIGATOARE

- Navigarea printre paginile web utilizând adrese concrete ale paginilor sau link-urile
- Schimbarea paginii de start
- Întoarcerea la o pagină accesată anterior
- Actualizarea paginii curente
- Memorarea adreselor de pagini web, în scopul accesării rapide a acestora
- Tipărirea la imprimantă a paginilor web
- Căutarea informațiilor
- Deschiderea mai multor pagini web simultan
- Descărcarea fișierelor din Internet
- Interpretarea codurilor client-side etc., etc.

EXAMPLE



MOTOARE DE CĂUTARE PE INTERNET

- Motorul de căutare pe Internet este un instrument de căutare a informațiilor pe WWW
- Informațiile căutate pot fi: pagini web, imagini, alte tipuri de fișiere
- Spre deosebire de directoarele web, care sunt administrate de oameni, motoarele de căutare operează automat, având la bază algoritmi de căutare. Uneori căutarea se face mixt: automat și cu ajutorul omului
- Un motor de căutare operează respectând următorii pași:
 - Navigare web
 - Indexare
 - Căutare

Suplimentar: *“Proiectarea, dezvoltarea și întreținerea siturilor web” de Nicolae Sfetcu sau vedeți SEO pentru diferite motoare de căutare*

ADRESE URL

- În cadrul serviciului WWW resursele rețelei Internet se specifică, prin intermediul unor adrese speciale, numite **adrese URL** (Uniform Resource Locator)
- Fiecare dispozitiv (calculator) primește, în afară de adresa numerică (o adresă unică primită de la furnizorul de Internet, denumită adresă Internet sau adresă IP), *un nume logic* (**hostname**)
- Sintaxa URL se bazează pe identificatorii mașinilor (IP și nume logice), pe localizarea fizică a resurselor (ca de exemplu drumurile de acces la fișiere) și pe protocoalele Internet existente
- Un URL identifică, în mod unic, o resursă de pe Internet, de exemplu o pagină web, un server, o imagine etc.
- URL-ul ajută utilizatorul să localizeze resursa Internet
- **URL-ul este alcătuit din litere, numere și semne de punctuație**

STRUCTURA URL

- Structura URL este ierarhică. Exemplu de URL:

Nume_protocol://nume_logic - poate fi urmat, în anumite cazuri, de nume de fișiere

- **Protocolul** este de obicei http (Hypertext Transfer Protocol) – protocolul de transfer a hipertextelor, dar el poate fi și FTP (File Transfer Protocol) sau

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

- Diferente:

<https://www.keycdn.com/blog/difference-between-http-and-https/>

- Numele logic este constituit conform unei scheme, ca regulă, cu 3 nivele ierarhice (de la dreapta la stânga)
- Primul nivel suporta initial 7 domenii internaționale (vezi: [RFC 920](#)), a câte 3 litere, regrupând mașinile care aparțin unor rețele internaționale (.com, .int, .net, .org) sau rețelelor care depind de administrația americană (.edu, .gov, .mil)
- Astazi sunt 6 nume care nu au limite la inregistrare:
.com, .net, .org, .info, .biz, .name

Lista completa: <http://www.iana.org/domains/root/db>

STRUCTURA URL. II

Rețelele naționale sunt adresate prin domenii codificate cu 2 litere (codul ISO al țării) - tot de nivel 1

.be *Belgia*

.ca *Canada*

.ch *Elveția*

.fr *Franța*

.jp *Japonia*

.md *Moldova*

.us *Statele Unite*

- În interior, domeniile sunt organizate în subdomenii (nivelul 2) care corespund unor entități economice. De exemplu, în domeniul **.com** se găsesc numeroase întreprinderi informatice (Microsoft, Apple, IBM ...), fiecare posedând propriul său subdomeniu (*usm.md*)
- Al treilea nivel îl constituie, ca regulă, mașinile (calculatoare, rutere etc.), care au toate un nume logic (<http://moodle.usm.md/>)
- Pentru a identifica o mașină prin numele său logic, cele 3 nivele sunt despărțite prin puncte, în ordinea *nume.subdomeniu.domeniu* ca în ex. ftp.microsoft.com
- Pentru o clasificare mai fină se pot adăuga alte nivele

PROTOCOLUL HTTP

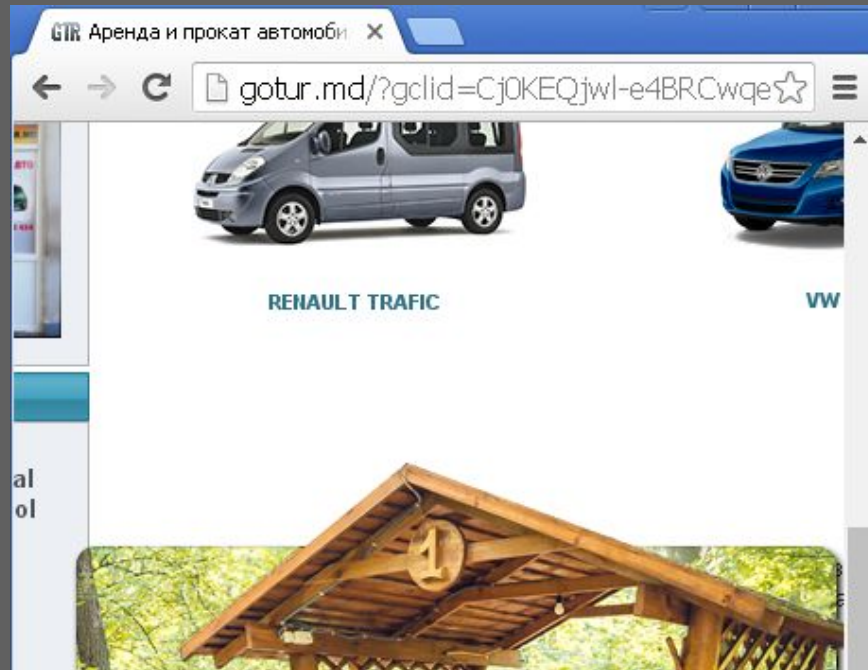
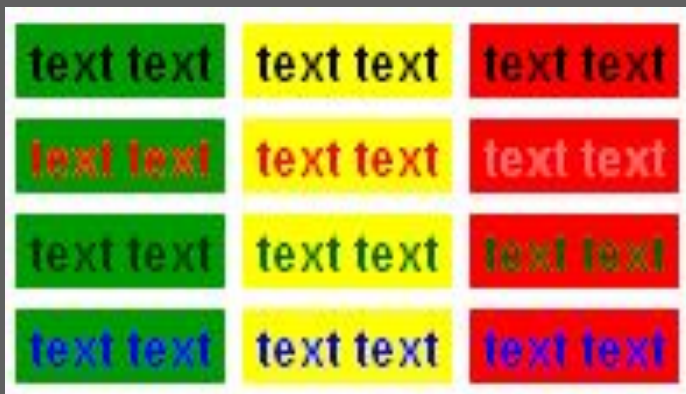
- HTTP este un protocol de bază în procesul transmiterii datelor prin WWW
- Se folosește pentru transmiterea hipertextului, a paginilor web de pe un calculator pe altul începând cu anul 1990
- Detalii vezi în standardul **RFC2068**

DEZVOLTAREA SITE-URILOR WEB

WEB SITE, WEB PAGE

- **Pagină web** — un document sau o resursă informațională, ce aparține serviciului web, și accesul la care se realizează prin intermediul unui browser web
- O pagină web tipică este un fișier textual, scrisă în format HTML, care poate conține referințe la fișiere de alt format precum: text, imagine grafică, video, audio, aplicații, servicii web și altele. Suplimentar, aceste fișiere pot conține referințe de tip hipertext pentru a realiza trecerea mai rapidă la alte pagini web sau fișiere
 - **Pagini web dinamice** — acea pagină web care este generată prin intermediul unui script, ca regulă server-side. Serverul generează pagina, iar browserul o prezintă utilizatorului
 - **Pagini web statice** — este un simplu fișier, care este stocat pe server, și la cerere este prezentat utilizatorului fără a fi modificat conținutul paginilor
- **Site, sau Web-Site** — una sau mai multe pagini web, între care sunt stabilite relații logice

DEFICIENȚE!

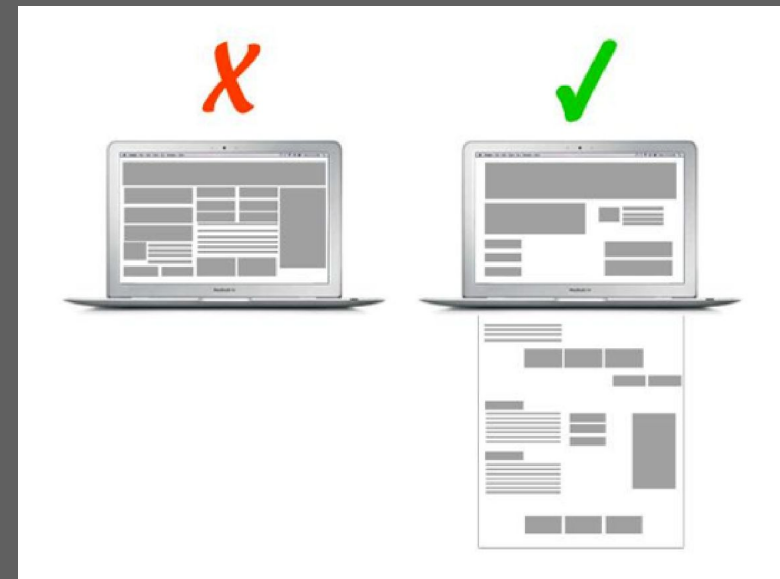




- Interfața grafică destinată utilizatorului:
- Plăcută ochiului
 - Comodă în utilizare
 - Adaptabilă dispozitivului de ieșire (responsive)

PE SCURT DESPRE - UI ȘI UX

- Interfața web destinată utilizatorului trebuie să fie
 - **Plăcută ochiului** (UI): se alege corect fundalul, iar pentru evidențierea unui text sau ale altor detalii în conținut, vor fi utilizate maximum 2-3 culori
 - **Comodă** (UX): produsul web dezvoltat, trebuie să fie la maximum comod și ușor de accesat de orice utilizator



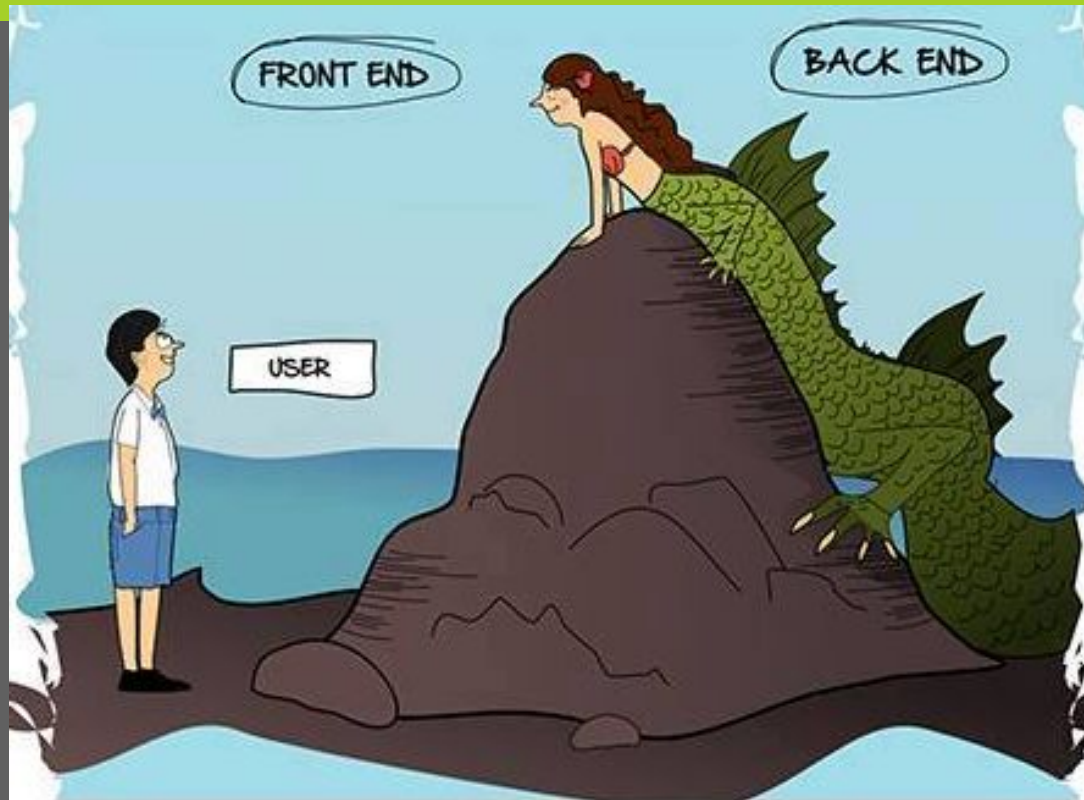
Selectati un set de culori:

https://www.shutterstock.com/blog/101-color-combinations-design-inspiration?kw=&gclid=CjoKCOjw1dGJBhD4ARIsANb6OdnxtZmCAFSGrzYHFm2iJXXyqHPsN1yT8b1cfas1YIn5fsOSrgyBwoaAhkTEALw_wcB

MODALITĂȚI ELABORARE SITE

Site-ul poate fi dezvoltat

- **Scriind codurile, toate, de la o, pentru întreg site-ul**
- Folosind, pentru unele elemente ale site-ului, framework-uri, iar pentru altele scriind propriile coduri
 - Exemple framework-uri pentru front-end: **w3.css**, **Bootstrap** etc.
- Folosind șabloane, care apoi pot fi adaptate, conform necesităților
 - Exemple de șabloane: http://www.w3schools.com/css/css_rwd_templates.asp
- Folosind site-uri specializate, care oferă posibilitatea dezvoltării site-ului on-line, de la distanță, utilizatorul nescriind în acest caz nici un cod. Astfel de site-uri sunt orientate la persoane care nu sunt din domeniul IT, dar doresc să-și facă un site de promovare, de exemplu
 - Exemplu platformă de creare: <http://fr.wix.com/>



Front end vs. Back end.

ELEMENTE NECESARE DEZVOLTĂRII FRONT-END-ULUI SITE-ULUI

- Dezvoltarea front-end-ului presupune elaborarea părții-client a site-ului. Persoana responsabilă de dezvoltarea front-end-ului elaborează o machetă a site-ului și realizează interfața destinată utilizatorului
- Pentru realizarea interfeței-web destinată utilizatorului se folosește HTML, CSS și JavaScript (sau frameworkuri, biblioteci bazate pe aceste tehnologii)
 - HTML este responsabil de definirea structurii și conținutului fiecărei pagini / componente a site-ului
 - CSS este responsabil de definirea aspectului fiecărei pagini / componente a site-ului
 - JavaScript este utilizat pentru a defini comportamentul paginilor / elementelor site-ului

ISTORIA DEZVOLTĂRII HTML

- HTML (Hyper Text Markup Language) - un limbaj utilizat pentru descrierea paginilor web
- HTML nu este un limbaj de programare, ci unul de marcare
 - Un **limbaj de marcare** (sau etichetare) reprezintă o mulțime de etichete / tag-uri de marcare
 - Limbajul HTML utilizează set-ul de tag-uri / de marcaje pentru a descrie paginile web
 - Un **tag** este un cuvânt sau o scurtă descriere care etichetează o informație, ușurând regăsirea acesteia
 - Tag-urile de marcare HTML se numesc simplu tag-uri HTML
- **Un document HTML** descrie o pagina web și conține tag-uri HTML și text simplu
- Un browser nu afișează tag-urile HTML, dar utilizează tag-urile dintr-un document HTML pentru a le interpreta și a genera conținutul unei pagini web, pe care o afișează într-o formă înțeleasă de om

CREAREA DOCUMENTELOR HTML

- Pentru a crea documente HTML este suficient cel mai simplu editor de texte
 - de exemplu Note-pad sau Note-pad++, Visual Studio Code, Sublime etc. - care este comod și simplu de utilizat în descrierea paginilor web
 - Toate aceste editoare trebuie instalate preventiv pe PC, pentru a fi posibilă descrierea paginilor web
- Va recomand să instalați pe calculatorul dvoastră VS Code, iar drept browser recomandat - GoogleChrome

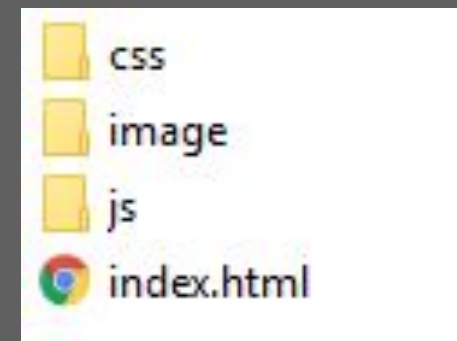
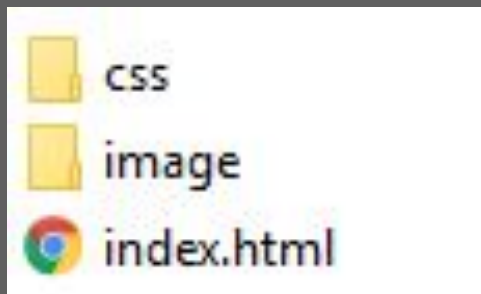
PAȘII... PENTRU GENERAREA UNEI PAGINI WEB

1. Se deschide editorul (de exemplu VS Code sau NotePad++)
2. Se creează documentul HTML
3. Se salvează documentul specificând numele și extensia
 - Orice fișier care descrie conținutul unei pagini web are un nume și o extensie
 - Extensia caracterizează tipul fișierului. Fișierele care descriu documente HTML au extensia *.html* sau *.htm*
 - Fie am creat fișierul *index.html*
4. (Se închide fișierul)
5. Se lansează browserul și în bara de adrese se scrie numele fișierului html, specificând și calea către fișier
 - Pentru vizualizarea rezultatului descrierii paginilor web este suficient un navigator Internet (browser – care "știe" să interpreteze codurile HTML)
 - Pentru validarea corectitudinii scrierii documentului web poate fi folosit un validator on-line:
<http://validator.w3.org/nu/>: <http://validator.w3.org/nu/> sau http://www.w3schools.com/html/html_validate.html

RECOMANDĂRI

- Se recomandă, **pentru un site**
 - Să se creeze o mapă separată, corespunzătoare numelui proiectului
 - În mapa respectivă să fie create minimum alte 2 mape: una rezervată pentru imaginile necesare site-ului și una necesară stilurilor folosite în site
 - De exemplu, proiectul meu "Donuts" este structurat astfel:

În cazul în care sunt folosite JavaScript-uri externe, creați o mapă JS sau JavaScript, în care veți păstra scripturile-client



VERSIUNI HTML

Versiunea	Anul apariției
HTML	1991
HTML+	1993
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2014
HTML5.1	2016, la nivel de recomandari

ELEMENTE HTML

- Un document HTML se definește folosind **elemente HTML**
- Elementele HTML sunt formate dintr-un tag de deschidere (tag de început) și un tag de închidere (tag de sfârșit)
- Între tag-ul de început și cel de sfârșit se amplasează conținutul elementului HTML
- Exemplu: `<tag_început attribute> Conținut </tag_sfârșit>`
- Unele elemente HTML pot să nu aibă conținut
 - Exemplu de element vid: eticheta pentru trecerea din rând nou `
`
 - Elementele vide se închid chiar în tag-ul de început (obligatoriu în versiunea XHTML): `
`

ELEMENTE HTML

- Denumirile tuturor tag-urilor sunt incluse între <>, tag-ul de sfârșit având după semnul < simbolul /
 - `<p> Salut!</p>` - exemplu de tag cu conținut
 - `
` - exemplu tag fără conținut
- Unele browsere afișează conținutul chiar dacă lipsește tag-ul de închidere, dar nu toate!!!
- Ca să nu aveți surprize la interpretarea codurilor HTML, străduiți-vă ca toate elementele să fie în pereche – tag de deschidere și tag de închidere
- Denumirile tag-urilor HTML nu sunt case-senzitive!
- În versiunea HTML5 au apărut noi elemente din domeniul multimedia:
 - pentru inserarea sunetelor, secvențelor audio, video
 - elemente noi specifice controalelor din formularele HTML și **elemente utilizate în definirea structurii paginii web**
- De asemenea din versiunea 5 au dispărut unele elemente prezente în versiunea 4.01

STRUCTURA DE BAZĂ A DOCUMENTULUI HTML

```
<html>  
  
  <head>  
    <title> Titlu paginii </title>  
  </head>  
  
  <body>  
    <h1> Un titlu </h1>  
    <p> Un paragraf </p>  
    <p> Inca un paragraf </p>  
  </body>  
  
</html>
```

Cea mai simplă pagină web constă din minimum 3 elemente

STRUCTURA DE BAZĂ A DOCUMENTULUI HTML. II

- Cea mai simplă pagină web constă din minimum 3 elemente
 - **<html>** — care reprezintă un container pentru alte tag-uri. În el se includ elementele **<head>** și **<body>**
 - **<head>** are destinația definirii antetului paginii web. Conținutul acestui tag nu este prezentat în browser. Scopul acestuia este de a stoca informații necesare browserului
 - **<body>** este destinat pentru a stoca definirea conținutului paginii, acel conținut care se va afișa în fereastra browserului

ADĂUGĂRI ÎN STRUCTURA DE BAZĂ

- În ultimele versiuni HTML (după 4.01) în structura de bază a unui document HTML s-a adăugat și declarația **<!DOCTYPE>**, înaintea tag-ului *html*
- Această declarație este necesară browsere-lor pentru a afișa corect pagina web
- Ea specifică versiunea HTML-ului în care a fost definită pagina
- Nu este un tag! Și nu este CASE senzitiv!
- În versiunea 5 structura de bază va fi:

<!DOCTYPE html>

<html>

<head><title>Titlul documentului</title></head>

<body>

Conținutul documentului HTML

</body>

</html>

EXEMPLU

- Câte elemente HTML sunt în acest exemplu?

```
<!DOCTYPE html>  
<html>  
  <head>  
    <title>Exemplu</title>  
  </head>  
  <body>  
    <h1>Un prim exemplu</h1>  
    <p>Un text aleatoriu</p>  
  </body>  
</html>
```

ATRIBUTELE ELEMENTELOR HTML

- Un element HTML poate avea sau poate să nu posede atribute
- Sunt multe elemente HTML care posedă atribute sau pot fi formate din alte elemente HTML
 - Atributele furnizează informații suplimentare despre un element
- Atributele sunt specificate în tag-ul de început a elementului HTML
- Se recomandă următoarea sintaxă de utilizare:
denumire_atribut="valoare"
 - ``

Standardele recomandă utilizarea ghilimeleor la specificarea valorilor atributelor! Lipsa ghilimelelor poate conduce la apariția erorilor.

- De obicei sunt folosite ghilimele duble însă în unele cazuri, atunci când sunt necesare a fi folosite ghilimele în ghilimele, pot fi folosite și ghilimelele unare

ATRIBUTE GLOBALE

- Atributele globale sunt acele atribute care pot fi utilizate în orice element HTML
- Exemple de atribute globale:

Atributul	Descrierea
class	Specifică unul sau mai multe denumiri de clase pentru un element HTML (au apărut ca rezultat a necesității definirii stilurilor)
id	Specifică un id , unic pentru un element HTML (se utilizează în special cu JS)
style	Specifică stilurile in-line
title	Specifică informații suplimentare referitoare la un element HTML

Mai sunt și atribute globale care definesc lansarea evenimentelor – li se pune în corespondență un script

Detalii: http://www.w3schools.com/tags/ref_eventattributes.asp

TAG-UL HTML

- Tag-ul `<html>` are rolul de a le „spune” browserelor că urmează un document HTML
- Tag-ul `<html>` mai este numit și tag-rădăcină pentru un anumit document HTML
- Tag-ul `<html>` este un tag de tip container pentru restul elementelor HTML
- Doar declarația `<!DOCTYPE>` nu se conține în elementul **HTML** – în rest toate elementele unui document HTML se includ în elementul **html**
- ***Are tag de închidere!***
- Toate browsere web susțin acest element



- **3** noțiuni noi învățate azi
- **2** întrebări/neclarități care au apărut
- **1** sugestie pentru tema următoare

De citit:

<http://webblog.bestwebimage.ro/2014/06/ce-inseamna-front-end-si-ce-inseamna-back-end/>

Trenduri in web-design:

<https://www.wix.com/blog/2019/11/web-design-trends-2020/>