

TW - Curs 1

Tehnologii:

1. HTML - limbaj de marcare pentru realizarea paginilor web
2. CSS - limbaj de descriere a stilurilor
3. JavaScript - limbajul de scripting care permite interactiunea cu paginile web
4. DOM - modelul orientat pe obiecte al unei pagini web disponibil in JS
5. AJAX - colectie de tehnologii bazate pe JS folosite in dezvoltarea aplicatiilor web
6. Node.js - mediu de executie JS in afara unui browser

Resurse online:

1. www.w3schools.com
2. developer.mozilla.org/en-us/docs/Web
3. www.w3.org/standards/webdesign
4. www.tutorialsteacher.com/nodejs

Carti JS si Node:

drive.google.com/drive/folders/1uJsUzLkT4CCAxathxLrul7k9dE7U8Yc9

Internet:

- Internet = sistem de retele de calculatoare interconectate care folosesc protocolul TCP/IP
- Protocol (specificatie interna) = o multime de reguli si conventii care faciliteaza si controleaza schimbul de date intre dispozitive diferite
- TCP/IP = stabileste si controleaza transmitia datelor (Orice echipament conectat la internet are o asignat o singura adresa IP, eticheta numerica)

World wide web (WWW, Web)

- Hypertext = text afisat pe un dispozitiv electronic care contine referinte (hyperlink) la date ce pot fi accesate imediat
- WWW = un sistem de documente hypertext interconectate care poate fi accesat in internet
- Functionalitatea se bazeaza pe:
 - HTTPS (HyperText Transfer Protocol + Securitate) = regulile de comunicare intre client si server
 - HTML (HyperText Markup Language) = creaza continutul documentelor web
 - URL (Uniform Resource Locator) = identifica unic paginile de web
- HTTP Request (Sintaxa unei cereri HTTP)

```
Method /path-to-source HTTP/version number
```

```
Header-Name-1: values
```

```
Header-Name-2: value
```

```
[ optional request body ]
```

- HTTP Request (Exemplu pentru adresa <https://developer.mozilla.org/en-US/search?q=www>)

```
GET /... search?q=www HTTP/2
```

```
Host: developer.mozilla.org
```

```
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:85.0)
```

```
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,
```

```
Accept-Language: ro-RO,ro;q=0.8,en-US;q=0.6,en-GB;q=0.4,en;q=
```

```
Accept-Encoding: gzip,deflate, br
```

```
Connection: keep-alive
```

- Metode HTTP:

```
GET - solicita un document resursa de pe server
```

```
HEAD - solicita informatii despre un document/resursa
```

```
POST - metoda utilizata pentru a transmite date catre server
```

```
PUT - metoda utilizata pentru a actualiza/depone o resursa pe
```

```
DELETE - metoda utilizata pentru a sterge un document/resursa
```

- Traditional, browserul web permite doar folosirea metodelor GET si POST
 - o metoda este sigura (safe) cand nu produce modificari in datele serverului
 - GET si HEAD sunt safe
 - POST, PUT, DELETE nu sunt safe
- HTTP Response (Sintaxa)

```
HTTP/version-number status-code message
```

```
Header-Name-1: value
```

```
Header-Name-2: value
```

```
[response body]
```

- HTTP Response (Exemplu pentru aceeași adresă de mai sus)

```
HTTP/2 200 OK
```

```
Date: Mon, 15 Feb 2021 08:01:28 GMT
```

```
Server: unicorn/20.0.4
```

```
Content-Type: text/html; charset=utf-8
Content-language: en-US
```

- Coduri de stare
 - 200 = OK
 - 301 = Moved Permanently
 - 400 = Bad Request
 - 401 = Unauthorized
 - 403 = Forbidden
 - 404 = Not Found
 - 500 = Internal Server Error
 - 503 = Service Unavailable
- Ce este un website?
 - o colectie de pagini web dar si de alte resurse (imagini, videoclipuri, fisiere de date, etc) publicate online sub acelasi nume de domeniu

Front-end	Back-end
Partea care interactioneaza cu utilizatorul (interfata grafica)	Partea care se ocupa de prelucrarea cererilor utilizatorilor
este controlata de browser	Poate comunica cu baza de date in vederea accesarii resurselor (creare, citire, stergere)
tehnologii implicate (HTML, CSS, JS, framework-uri)	tehnologii: JS la nivel de server

- HTML (HyperText Markup Language)
 - HTML este un limbaj de marcare folosit pentru a defini structura si a descrie continutul unei pagini web (NU este lb de programare)

- Documentele HTML contin elemente (taguri) care descriu diferite tipuri de continut: paragrafe, ancore, liste, imagini, tabele, video, etc.
- Browserele interpreteaza elementele si afiseaza documentele HTML intr-o forma grafica.
- Un document HTML:
 - Poate fi scris folosind orice editor simplu de texte (ex. Notepad++) sau editoare mai complexe (ex. VS Code, Brackets, Adobe Dreamweaver)
 - se salveaza cu extensia htm sau html.

- Element (tag) HTML:

- sintaxa:

```
<tag> continut </tag>
```

- tagurile HTML nu sunt case sensitive
- exista 2 tipuri de elemente:

elemente cu continut:

```
<p> Acesta este un paragraf </p>
```

elemente fara continut:

```
<br>, <hr>, <img>, <link>, etc.
```

- HTML 5 (Noutati):

- Imbunatateste continutul semantic prin taguri noi (<main>, <section>, <article>, <header>, <footer>, <aside>, <figure>, <figcaption>, etc).
- adauga taguri multimedia (<audio>, <video>)
- adauga taguri pentru grafica (<svg>, <canvas>)

- Structura unui document HTML:

```
<!DOCTYPE html> <!-- Obligativu HTML5 -->
```

```

<html lang="ro">

<head>
<title> Titlul Paginii </title>
</head>

<body>
<h1> O pagina simpla </h1>
<p> Aceasta este o pagina <a href ="http://dexonline.ro/d
<!-- comentariu -->

</html>

```

- Elemente block:

- elementele block incep si se termina cu o linie noua si ocupa intreaga latime a elementului parinte
- pot contine elemente block sau inline
- in versiunea HTML5 corespund categoriei flow content
- ex: <p>, <div>, <h1>, <h6>, etc
- ex cod (<p> este un element block) :

```

<style>
div {border: 1px solid black;
      width: 200px;
      height: 100px;
    }
p {background-color: yellow;}
</style>

<head>
<body>

<div>

```

```
<p>Primul Paragraf</p>
<p>Al doilea Paragraf</p>
</div>
...
```

- Elemente inline:

- elementele inline ocupa doar spatiul rezervat elementului, fara a ocupa toata latimea parintelui.
- nu incep neaparat cu o linie noua
- pot contine alte elemente inline
- in versiunea HTML5 apartin categoriei phrasing content
- ex: , , <a>
- ex cod:

```
<style>
span{background-color: yellow;}
</style>

<head>
<body>

<div>
<p>Acesta este <span>un paragraf</span> important</p>
</div>
...
```

- Atribute

- nume-atribut = "valoare"
- <tag atribut1 = "valoare1" atribut2="valoare 2"...>
- globale (se pot asocia tuturor elementelor)
 - ex: lang, id, class

```
<p id="p1" class="abc">...</p>
```

- specifice unui anume element
 - ex: href → a, link ; src →img, frame ; type, start → ol

```

```

- Elemente semantice:

- Beneficii:
 - structura mai clara
 - evaluare optima pentru motoarele de cautare (SEO)
 - paginile web pot fi mai bine intelese de anumite aplicatii (e.g text-to-speech)

```
<!--! Ex structura semantica -->
<header></header>
  <section>
    <article>
      <figure>
        <img>

<figcaption></figcaption>
      </figure>
    </article>
  </section>
<footer></footer>

<!--! Ex structura non-semantica -->
<div id="header"></div>
  <div class="section">
    <div class="article">
      <div class="figure">
        <img>
```



```
        <div class="figcaption"></div>
    </div>
</div>
</div>
<div id="footer"></div>
```