

Curs 3

2023/2024

Programarea aplicațiilor web

- Programarea aplicațiilor web
 - An V RC
 - 1.5C/1L/1P

Program

- An V
 - Saptamana 1
 - Luni 17-20 Curs (Intro/HTML/CSS)
 - Saptamanile 2-8
 - Luni 16:30-18 Curs
 - Luni 18-20 Laborator
 - Saptamanile 9-14
 - Luni 16:30-18 Curs
 - Luni 18-20 Proiect

Orar

- <https://orar.etti.tuiasi.ro/> : C->16:30-18, L/P -> 18

Group: Professor: Classroom:



FACULTATEA DE ELECTRONICA, TELECOMUNICATII SI TEHNOLOGIA INFORMATIEI

55RC

ETTI_

	1 8:00 - 8:50	2 9:00 - 9:50	3 10:00 - 10:50	4 11:00 - 11:50	5 12:00 - 12:50	6 13:00 - 13:50	7 14:00 - 14:50	8 15:00 - 15:50	9 16:00 - 16:50	10 17:00 - 17:50	11 18:00 - 18:50	12 19:00 - 19:50
L												
Ma												
Mi												
J												
V												
Sa												

Nota

- An V
 - 33% E
 - 66% Aplicatii
 - 33% L
 - 33% P

Site



Microwave and Optoelectronics Laboratory



We are enlisted in the Telecommunications Department of the Electronics, Telecommunication and Information Technology Faculty (ETTI) from the "Gh. Asachi" Technical University (TUIASI) in Iasi, Romania

We currently cover inside ETTI the fields related to:

- Microwave Circuits and Devices
- Optoelectronics
- Information Technology

Courses

Nr.	Course	Shortcut	Code	Type	Semester	Credits	Weekly	Examination	Link
1	Microwave Devices and Circuits for Radiocommunications	DCMR	DOS412T	DOS	7	4	0P,1L,0S,2C	Exam	
2	Monolithic Microwave Integrated Circuits	CIMM	RD.IA.207	DOMS	11	6	1.5L,0S,2C,0P	Exam	
3	Advanced Techniques in the Design of the Radio-communications Systems	TAPSR	RD.IA.103	DIMS	9	6	1.5P,0L,0S,2C	Exam	
4	Optical Communications	CO	DOS409T	DOS	7	5	0P,1L,0S,3C	Colloquiu	
5	Optical Communications	OC	EDOS409T	DOS	7	5	0P,1L,0S,3C	Exam	
6	Satellite Communications	CS	RC.IA.104	DIMS	9	6	0L,0S,2C,1.5P	Exam	
7	Applied Informatics 1	IA1	DOF135	DOF	1	4	0P,1L,0S,2C	Verificati	
8	Applied Informatics 1	AI1	EDOF135	DOF	1	4	0P,1L,0S,2C	Verificati	
9	Databases, Web Programming and Interfacing	DWPI	ITT.IA.601	DIS	11	5	1P,1L,0.25S,1C	Verificati	
10	Web Applications Design	PAW	RC.IA.108	DIMS	10	5	1L,0S,1.5C,1P	Exam	
11	Optoelectronics	OPTO	DID405M	DID	8	4	0P,1L,0S,2C	Colloquiu	
12	Microwave Devices and Circuits for Radiocommunications (English)	MDCR	EDOS412T	DOS	8	4	0P,1L,0S,2C	Exam	




Curs

- Curs strict orientat spre laborator/proiect
 - se predă ceea ce se folosește în aceeași zi la laborator/proiect
 - curs exact înainte de laborator/proiect
- Orientat spre practică
 - 90% practică
 - 10% teorie
- Memorarea lui inutilă la examen
- Prezența
 - 3pz = 0.5p Examen (maxim 2p)


Proiect

- Teme in **echipa**: 2/3 membri
- Evaluare **individuala**
- Variabile ca dificultate (cu note diferite)

Notare 2024 (preconizat – Sg final)

- 1p – functionalitate
- 1p – mutarea **personală** a site-ului (restaurare backup + setare server) pe un server de referinta **Ubuntu/Debian** 
- 1p – cunoasterea codului
 - raspunsul la intrebari de genul: "unde ai facut aceasta", "ce face acest cod"
- 1p – planul aplicatiei
- Teme "de nota 10,10+"
 - Initiativa. Investigarea posibilitatilor de imbunatatire
 - Explicatii relativ la functionarea unei anumite secvente de cod
 - Utilizare sesiune, Javascript, **CSS media**

Documentatie

- RF-OPTO
 - <http://rf-opto.etti.tuiasi.ro>
- Fotografie
 - ~~de trimis prin email:~~ [examen online](#)
 - necesara la laborator/curs
 - +1p ex. (S4), +0.5p ex (S6) 
- Adresa de email
 - necesara pentru accesul la zonele sensibile de pe server
 - **obligatorie** pentru activitatea **online**
- Exemplu de teme si specificatii detaliate pentru proiect

Server referinta LAMP 2024

- 3 variante acceptate
 - CentOS 7.1
 - Ubuntu 20.04
 - Debian 12.5

Tema bonus

- logfile.php
 - Afiseaza log Apache (erori php majore)
- **2p suplimentar** la laborator/examen
- Modificare logfile.php pentru a afisa **toate** erorile PHP
 - php.ini – activare erori
 - php.ini – locatie erori
 - logfile.php – afisare log PHP

~Tema bonus Win2000

- se verifica in php.ini ca erorile sunt activate
 - `error_reporting = E_ALL` (linia 349)
- se activeaza in php scrierea erorilor in fisier
 - `error_log = "C:\PHP\php.log"` (linia 412)
- se recupereaza logfile.php din sever-ul CentOS sau Ubuntu
- se schimba locatia log-ului Apache
 - `$log_file= 'C:\Program Files\Apache Group\Apache2\logs\error.log';`
(linia 53)
- se adauga dupa exemplul Apache afisarea log-ului PHP
 - `$log_file_php= 'C:\PHP\php.log';`
 - ... + utilizarea aceluui fisier (probabil **mai tarziu** in semestru)
- **Indeplinirea temei bonus** pe unul din server-ele de referinta necesita investigari mai detaliate legate de locatii/drepturi de acces pe sistemele Linux!!!

CURS

I.	HTML si XHTML (recapitulare)	1 oră
II	CSS	2 ore
III	Baze de date, punct de vedere practic	1 oră
IV	Limbajul de interogare SQL	4 ore
V	PHP - HyperText Preprocessor	8 ore
VI	XML - Extended Mark-up Language si aplicatii	4 ore
VII	Conlucrare intre PHP/MySql, PHP/XML, Javascript/HTML	2 ore
VIII	Exemple de aplicatii	6 ore
	Total	28 ore

LABORATOR

I.	Implementarea unui sistem de dezvoltare a aplicatiilor Web, instalare PHP, MySql, Apache si legaturile dintre ele	2 ore
II	Design web avansat folosind CSS	2 ore
III	Interogarea unei baze de date. Exercitii SQL	2 ore
IV	Utilizare PHP I	2 ore
V	Utilizare PHP II	2 ore
VI	Utilizare PHP pentru accesul la o baza de date	2 ore
VII	Aplicatie distribuita complexa	2 ore
	Total	14 ore

HTML

Forme

Aplicatie Web

- presupune prelucrarea unor date si oferirea unui document personalizat (rezultat al datelor respective)
- datele pot fi obtinute:
 - de la utilizator
 - o sursa externa (baze de date)
 - **combinatie** utilizator/baze de date

Forme in HTML

- necesare pentru ca utilizatorul sa poate trimite date server-ului
- **<form>...</form>**
- Attribute specifice:
 - action: adresa documentului care preia datele
 - `<form action="<?php echo $_SERVER['PHP_SELF'];?>">`
 - `<form action="actiuni/fisier2.html">`
 - method: modalitatea de transmitere a datelor: post sau get
 - `<form method="post" action= ... >`

Metode de transmitere

- **post** datele sunt transmise in bloc
- **get** datele sunt atasate adresei documentului de procesare : <results.php?prob=81&an=2009>
- **get** trebuie folosit numai cand datele sunt “idempotente”,
 - nu cauzeaza efecte colaterale
 - nu modifica starea server-ului (baze date, etc)
- se poate simula realizarea unei forme (**get**) prin scrierea corespunzatoare a link-urilor

Elemente de interactiune cu utilizatorul – <form>

- in interiorul etichetei <form>...</form>
 - input
 - select/option
 - textarea
 - button
- **toate** elementele vor avea un nume
 - atribut: name=""
 - numele va fi intalnit in uri-ul generat prin get, sau in numele variabilei trimisa prin post
- trimiterea datelor se face prin intermediul unui buton cu tipul type="submit"

INPUT

- `<input .../>`
- Attribute:
 - type: text | password | checkbox | radio | submit | reset | file | hidden | image | button
 - name: numele variabilei
 - value: valoarea trimisa server-ului la selectie (valoarea initiala in anumite cazuri)
 - checked/src/size/maxlength in functie de tip

Example

```
<input name="textfield" type="text" value="ceva" />
```

```
<input name="Ok" type="submit" value="Trimite" />
```

```
<label><input name="check" type="checkbox" value="5" checked />check1</label>
```

```
<label><input name="RG1" type="radio" value="a" checked="checked" />but1</label>
```

```
<label><input name="RG1" type="radio" value="b" />but2</label>
```

```
<input name="hid" type="hidden" value="6" />
```

ceva

Trimite



check1



but1



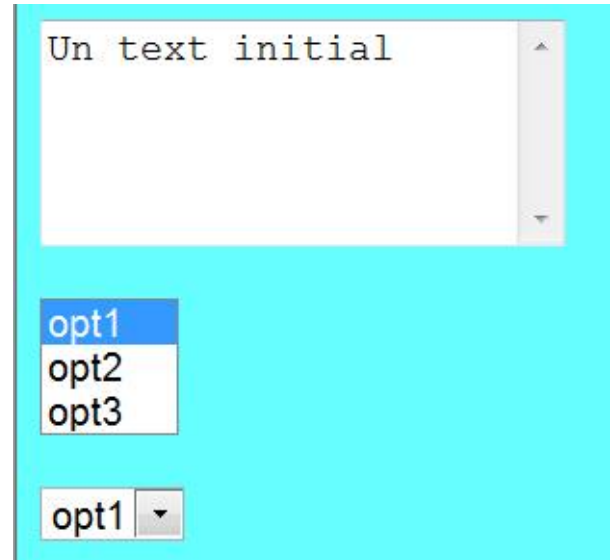
but2

TEXTAREA/SELECT

```
<textarea name="textarea" cols="20" rows="5">Un text  
initial</textarea><br /><br />
```

```
<select name="select." size="3">  
  <option value="1" selected="selected">opt1</option>  
  <option value="2">opt2</option>  
  <option value="3">opt3</option>  
</select><br /><br />
```

```
<select name="select...">  
  <option value="1" selected="selected">opt1</option>  
  <option value="2">opt2</option>  
  <option value="3">opt3</option>  
</select>
```



Un text initial

opt1
opt2
opt3

opt1 ▼

BUTTON

```
<button type="button" onClick="do ( );">Click Here</button>
```

```
<button type="submit" value="infoOnly">Request Info</button>
```

```
<button type="reset">Clear</button>
```

```
<button type="submit" id="sender"  
value="infoOnly">Request<br />Info<br /></button>
```

```
<button type="submit" id="sender"  
value="infoOnly"><p>Request</p><p><b><i>Info</i></b></p>  
</button>
```

- introdus pentru a oferi posibilitatea introducerii de continut mai complex (text formatat, imagini) in interiorul butoanelor de pe forma

Click Here

Request Info 

 Clear

Request
Info



Request

Info



HTML 5 – Validare input

- `<input type="email">`
- `<input type="number">`
- `<input type="search">`
- `<input type="tel">`
- `<input type="url">`

HTML 5 – Validare input

```
<label for="email">Email:</label>
<input type="email" id="email" name="email">
<br /><br />
<label for="can">Cantitate (1 - 5):</label>
<input type="number" id="can" name="can" min="1" max="5">
<br /><br />
<label for="gsearch">Cautare Google:</label>
<input type="search" id="gsearch" name="gsearch">
<br /><br />
<label for="phone">Telefon:</label>
<input type="tel" id="phone" name="phone"
placeholder="0123.456.789" pattern="[0-9]{4}.[0-9]{3}.[0-9]{3}"
required>
<br /><br />
<label for="server">Server:</label>
<input type="url" id="server" name="server">
```

Email:

Cantitate (1 - 5):

Cautare Google:

Telefon:

Server:

HTML 5 – Validare input

Email:

Please include an '@' in the email address. 'ceva' is missing an '@'.

ceva@server

Cantitate (1 - 5):

Telefon:



Please match the requested format.

Server:



Please enter a URL.

Email:



Please include an '@' in the email address. 'ceva' is missing an '@'.

Telefon:

Server:



Please fill out this field.

Telefon:

S



Please match the requested format.

Server:

HTML 5 – Interfata nativa browser

- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="month">`
- `<input type="range">`
- `<input type="time">`
- `<input type="week">`

HTML 5 – Interfata nativa browser

```
<label for="culoare">Culoare:</label>
<input type="color" id="culoare" name="culoare" value="#ff0000">
<br /><br />
<label for="nastere">Data nastere:</label>
<input type="date" id="nastere" name="nastere">
<br /><br />
<label for="examen">Examen (data si ora):</label>
<input type="datetime-local" id="examen" name="examen">
<br /><br />
<label for="fin">Finalizare (luna si an):</label>
<input type="month" id="fin" name="fin">
<br /><br />
<label for="vol">Volum (0 - 50):</label>
<input type="range" id="vol" name="vol" min="0" max="50">
<br /><br />
<label for="ora">Ora intalnire:</label>
<input type="time" id="ora" name="ora">
<br /><br />
<label for="sap">Saptamana:</label>
<input type="week" id="sap" name="sap">
```

Culoare:

Data nastere:

Examen (data si ora):

Finalizare (luna si an):

Volum (0 - 50):

Ora intalnire:

Saptamana:

HTML 5 – Interfata nativa browser

Culoare:

Data nastere:

Examen (data):

Finalizare (data):

Volum (0 - 50):

Ora intalniri:

Saptamana:

R G B ↕

Culoare:

Data nastere:

Examen (data):

Finalizare (data):

Volum (0 - 50):


Ora intalniri:



Saptamana:


March 2023

Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18

HTML 5 – Interfata nativa browser

Examen (data si ora): 03/13/2023 05:24 PM 

Finalizare (luna si an) March 2023  

Volum (0 - 50): 

Ora intalnire: --:--

Saptamana: Week --,

Trimite

Su	Mo	Tu	We	Th	Fr	Sa
26	27	28	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Clear Today

02 21 PM

03 22 AM


04 23


05 24

06 25

07 26

08

Finalizare (luna si an): June 2023 

Volum (0 - 50): 

Ora intalnire: --:--

Saptamana: Week --,

Trimite

2023

Jan Feb Mar Apr

May Jun Jul Aug

Sep Oct Nov Dec

Clear

This month

HTML 5 – Interfata nativa browser

Volum (0 - 50):



Ora intalnire: 05:30 PM

Saptamana: W

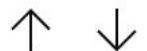
Trimite

02	25	PM
03	26	AM
04	27	
05	28	
06	29	
07	30	
08	31	

Saptamana: Week 13, 2023

Trimite

March 2023



Week	Su	Mo	Tu	We	Th	Fr	Sa
9	26	27	28	1	2	3	4
10	5	6	7	8	9	10	11
11	12	13	14	15	16	17	18
12	19	20	21	22	23	24	25
13	26	27	28	29	30	31	1
14	2	3	4	5	6	7	8

Clear

This week

Capitolul II

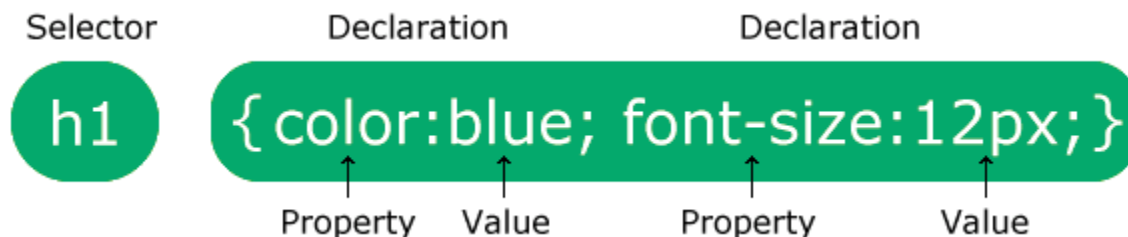
CSS

CURS

I.	HTML si XHTML (recapitulare)	1 oră
II	CSS	2 ore
III	Baze de date, punct de vedere practic	1 oră
IV	Limbajul de interogare SQL	4 ore
V	PHP - HyperText Preprocessor	8 ore
VI	XML - Extended Mark-up Language si aplicatii	4 ore
VII	Conlucrare intre PHP/MySql, PHP/XML, Javascript/HTML	2 ore
VIII	Exemple de aplicatii	6 ore
	Total	28 ore

CSS Concepte

- Specificatiile CSS sunt de forma:
 - **Selector { "regula CSS"; ["regula CSS";] }**
- Reguli CSS de forma:
 - atribut:valoare;
- Comentarii
 - Similar cu C/C++ , PHP, etc.
 - Cuprinse intre `"/*"` si `"*/"`



Selector CSS

- Selector HTML
 - conceput pentru a modifica modalitatea de afisare standard a etichetelor HTML
- Selector clasa
 - conceput pentru a lucra cu atributul HTML `class="..."`
 - util pentru aplicare uniforma a formei mai multor elemente din document
- Selector identificator
 - conceput pentru a lucra cu atributul HTML `id="..."`
 - util pentru aplicarea unei forme specifice unui anume element din document

Selector CSS

- Redefinirea etichetelor HTML standard
 - exemplu: afisarea implicita a tuturor celulelor de tabel din document este schimbata
- Redefinirea se poate face **si** doar in interiorul unei clase
 - exemplu: in interiorul claselor "footer" link-urile vor fi afisate altfel

```
td {  
  font-family:Tahoma, Verdana;  
  font-size:14px;  
  color:#D5A787;  
  vertical-align: top;  
  text-align: center;  
}
```

```
.footer a {  
  color:#FFD2B3;  
  text-decoration:none;  
}  
.footer a:hover {  
  color:#ffffff;  
  text-decoration:none;  
}
```

Selector CSS

- clase – “.”
 - Semnul “.” inaintea numelui
 - permit reutilizarea unei anumite forme grafice
 - `<p class="menu">...</p>`
 - `<h1 class="menu">...</h1>`
 - `<tr class="menu">...</tr>`
- identificador element – “#”
 - Semnul “#” inaintea numelui
 - `<tr class="menu">...`
`<td id="menu_curent">...</td>`
`...</tr>`

```
.menu {  
font-family:Tahoma, Verdana;  
font-size:14px;  
color:#6F3A15;  
text-align:center;  
font-weight:bold;  
}
```

```
#menu_curent {  
font-family:Tahoma, Verdana;  
font-size:15px;  
color:#FFFFFF;  
text-align:center;  
font-weight:bold;  
}
```

Selector CSS

- selector clasa si identificator concepute pentru a fi utilizate cu elementele de grupare HTML
 - `<div>...</div>` : grupare de tip bloc ("block")
 - `...` : grupare inserata ("in-line")
- anumite reguli au efect numai daca sunt atribuite unor elemente corespunzatoare (block sau in-line)

Specificari complexe ale selectorilor

- **.clasa1** {} – pentru orice element caruia i se atribuie atributul HTML class="clasa1"
- **p.clasa1** {} – numai paragrafelor care sunt afectate clasei "clasa1" <p class="clasa1">...
- **.clasa1 p** {} – oricarui paragraf aflat in interiorul unui bloc afectat clasei "clasa1"
 - <div class="clasa1"><p>...</p><p>...</p></div>
- **[attribute=value]** – oricarui element cu un anumit atribut si o anumita valoare
 - [target=_blank]
- in anumite conditii speciale (mai ales pentru eticheta HTML <a>)
 - **a:hover** {}, **a:visited** {}

Concepte CSS

CSS concepte

- Trei modalitati de a specifica stiluri
 - stiluri externe (External style sheet)
 - stiluri interne (Internal style sheet)
 - stiluri inserate (Inline style)
- Trei entitati care impun stiluri:
 - dispozitivul de afisare (browser)
 - autorul documentului
 - cititorul documentului

CSS concepte

- In orice moment un element din document poate fi sub controlul mai multor reguli care se contrazic
- Ordinea priorităților (in ordinea creșterii acesteia)
 - dispozitivul de afișare (browser)
 - autorul documentului – in ordinea definirii. De obicei:
 - External style sheet
 - Internal style sheet
 - Inline style
 - cititorul documentului (in diverse forme, "text size", "zoom" de ex)
- Prioritatea maxima data cititorului documentului este in dezbatere si inconsistentă (nu la toate regulile)

Ordinea prioritatilor CSS

High Priority	CSS Source Type	Description
1	Importance	The 'important' value overwrites the previous priority types
2	Inline	A style applied to an HTML element via HTML 'style' property
3	Media Type	A property definition applies to all media types, unless a media specific CSS defined
4	User defined	Most browsers have the accessibility feature: a user defined CSS
5	Selector specificity	A specific contextual selector (#heading p) overwrites generic definition
6	Rule order	Last rule declaration has a higher priority
7	Parent inheritance	If a property is not specified, it is inherited from a parent element
8	CSS property definition in HTML document	CSS rule or CSS inline style overwrites a default browser value
9	Browser default	The lowest priority: browser default value is determined by W3C initial value specifications

External style sheet

- regulile CSS se gasesc intr-un fisier extern, de obicei cu extensia “.css”
- fisierul respectiv este incarcat in document in sectiunea antet
 - `<head><link rel="stylesheet" type="text/css" href="stiluri.css" /></head>`
- utilizare obligatorie cand o suita de documente au o prezentare comuna

```
<head>  
<link rel="stylesheet" type="text/css" href="stiluri.css" />  
</head>
```

External style sheet

- Dezavantaje –minore in conditiile actuale
 - necesitatea unei conexiuni suplimentare la server
 - cantitatea de date mai mare (doar la prima pagina incarcata – se aduc si stilurile care nu sunt folosite)
- Avantaje
 - separa total **continutul** de **forma**
 - tratare unitara a paginilor, usureaza
 - navigarea, dpdv. al utilizatorului
 - intretinerea si modificarea intregului site, dpdv. al designer-ului
 - dimensiune minima a documentelor (o regula se aplica de multe ori dar se scrie o singura data)
 - incepand de la a doua pagina din suita, browser-ul este deja in posesia stilurilor, viteza e sporita

Internal style sheet

- Regulile CSS sunt indicate de asemenea in sectiunea antet (<head>... </head>)
- Se foloseste eticheta HTML <style>...</style>
- Se foloseste cand un document are o forma unica in site

```
<head>  
<style type="text/css">  
hr {color:sienna;}  
p {margin-left:20px;}  
body {background-image:url("b.gif");}  
</style>  
</head>
```

Internal style sheet

- Avantaj
 - se elimina conexiunea suplimentara la server
 - separa **continutul** de **forma**
- **Dezavantaj/Avantaj**
 - Daca in sectiunea antet apare dupa fisierul extern de stiluri, va suprascrie stilurile corespunzatoare
- Dezavantaj
 - nu permite tratarea uniforma a documentelor

Inline style sheet

- Se utilizeaza **atributul** HTML `style=""`
- Modifica o singura eticheta
- `<p style="color:sienna;margin-left:20px">Un paragraf</p>`
- De obicei utilizat punctual pentru a introduce efecte ce nu pot fi obtinute cu HTML
- De evitat:
 - **nu** separa forma de continut
 - creste dimensiunea documentului (vezi un fisier HTML salvat cu Microsoft Word)

```
<p style="color:sienna;margin-left:20px">Un paragraf</p>
```

Specificitati CSS (ponderi)

Selectors	Specificity
H1 {color: white;}	1
P EM {color: green;}	2
.grape {color: red;}	10
P.bright {color: blue;}	11
P.bright EM.dark {color: yellow;}	22
#id218 {color: brown;}	100
style=" "	1000

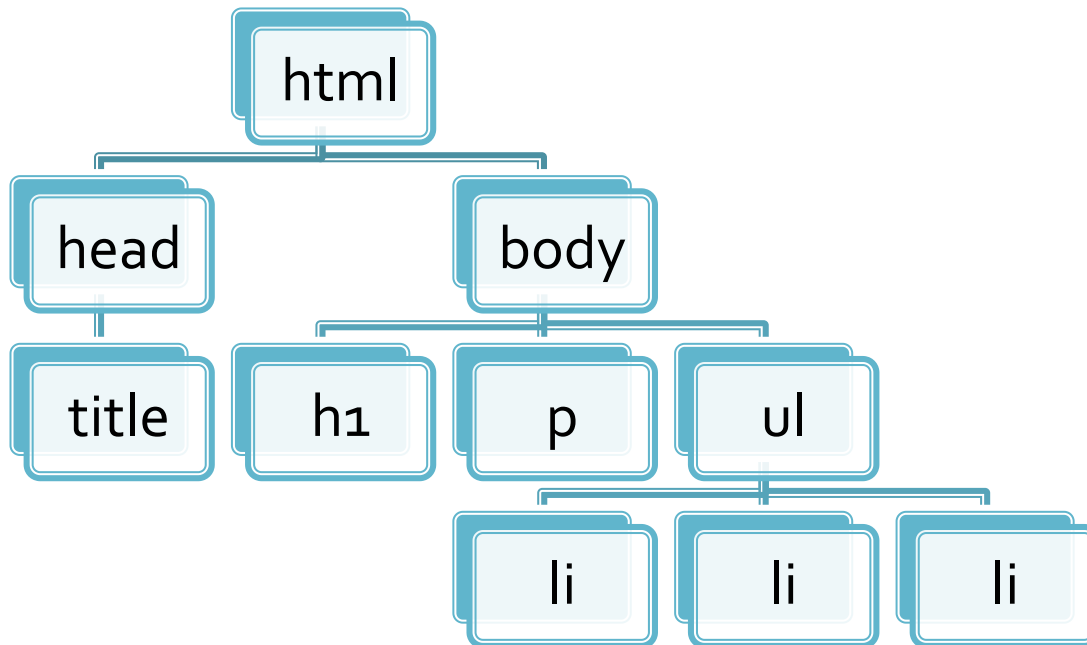
Exemplu specificitati

- `style="color: blue"`
- `suprascrie`
- `#xyz { color: red; }`

```
< <html>
<head>
<meta charset="utf-8" />
<style type="text/css">
#xyz { color: red; }
</style>
</head>
<body>
<p id="xyz" style="color: blue"> Demonstratie </p>
</body>
</html>
```

Relatia parinte copil si mostenire

- Orice document are o structura de tip graf in care elementele sunt introduse in interiorul altor elemente (DOM)



```
<html>
  <title>pagina mea</title>
  <body>
    <h1>Compozitori:</h1>
    <p>
    <ul>
      <li> elvis costello
      <li> johannes brahms
      <li> georges brassens
    </ul>
  </body>
</html>
```

Relatia parinte copil si mostenire

- in CSS valoarea implicita a multor reguli este **inherit** mostenire de la parinte
- Exemplu:
 - daca pentru body se fixeaza regula: `body {color:#FF0000;}` textul va fi implicit rosu pentru toate elementele incluse (p,li,ul) daca:
 - nu apar reguli suplimentare (p {color:black;})
 - nu exista reguli predefinite (h1 – regula browser-ului – h1 mai mare decat p, dar nu afecteaza culoarea)

```
<html>
<title>pagina mea</title>
<body>
  <h1>Compozitori:</h1>
  <p>
    <ul>
      <li> elvis costello
      <li> johannes brahms
      <li> georges brassens
    </ul>
  </p>
</body>
</html>
```

Elemente generale

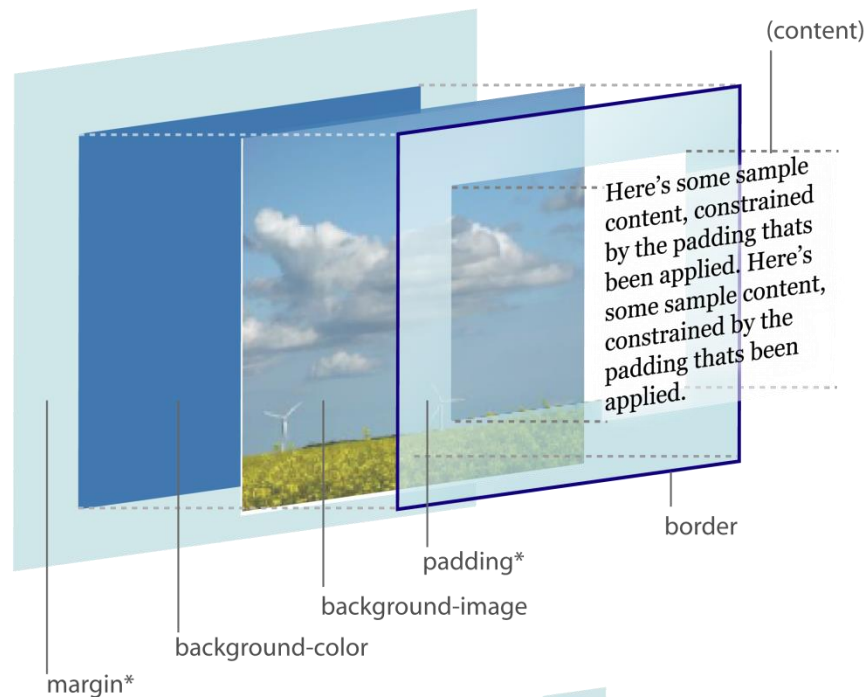
Dimensiunile in CSS

- relativ la dimensiunea implicita in browser
 - xx-small, x-small, small, medium, large, x-large, xx-large (implicit medium)
- relativ la dimensiunea din blocul parinte
 - smaller, larger
- absolute
 - in, cm, mm, pt (1 point = 1/72 in), pc (1 pica = 12 pt)
- relative
 - em, ex – relative la dimensiunea in blocul parinte
 - un cuvânt cu dimensiunea 2em = dimensiunea de 2 ori mai mare decat a textului din acel paragraf
 - precizie mai buna, valori fractionare permise (1.25em, 0.85em etc.)
 - px (pixeli) dependent de dispozitivul de afisare

CSS Box Model

- Orientat in jurul conceptului de “cutie” – Box model

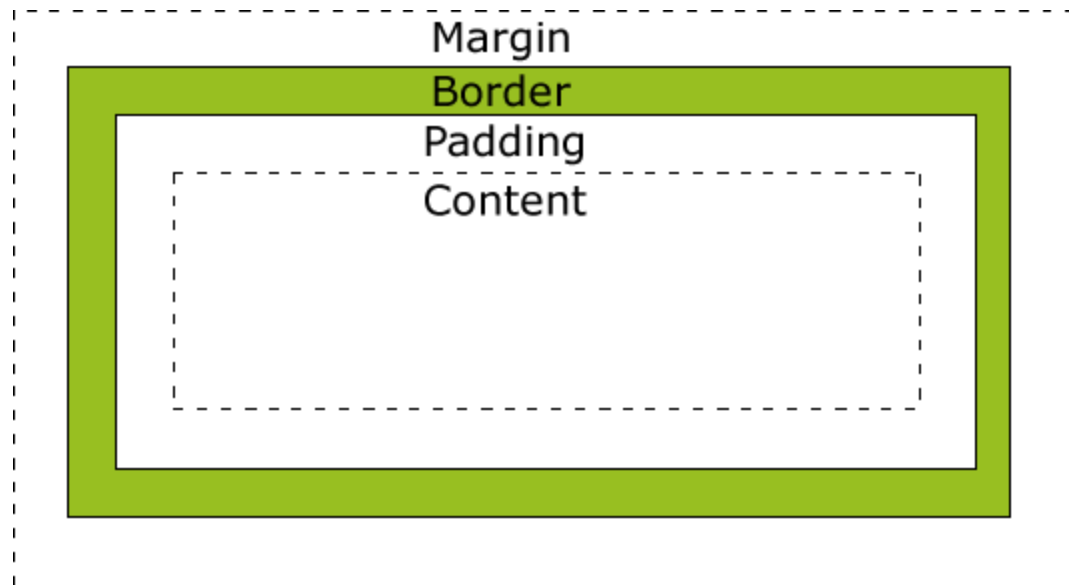
THE CSS BOX MODEL HIERARCHY



* transparent elements

CSS Box Model

- parametrii “box model” pot fi aplicati tuturor elementelor cu structura bloc, controland intregul bloc (margini, chenar, distanta intre chenar si continut).



Margine

- spatiu liber lasat in exteriorul blocului
- **margin-top**: [valoare] | auto;
- **margin-right**: [valoare] | auto;
- **margin-bottom**: [valoare] | auto;
- **margin-left**: [valoare] | auto;
- scurt
 - **margin**: [top] [right] [bottom] [left];
 - **margin**: [top] [right= left] [bottom];
 - **margin**: [top=bottom] [right= left];
 - **margin**: [top=bottom=right= left];

Chenar

- **border-style**:none | hidden | dotted | dashed | solid | double | groove | ridge | inset | outset | inherit; - deblocheaza celelalte proprietati
- **border-color**:culoare;
- **border-width**:thin | medium | thick | [valoare] | inherit;
- scurt: **border**: [border-width] [border-style] [border-color];
 - border:5px solid red;
- detaliat:
 - **border-left-color**:...
 - **border-top-width**...
 - etc.

Distanța chenar-continut

- spațiu liber lasat în interiorul blocului
- `padding-top:[valoare] | auto;`
- `padding-right:[valoare] | auto;`
- `padding-bottom:[valoare] | auto;`
- `padding-left:[valoare] | auto;`
- scurt
 - `padding: [top] [right] [bottom] [left];`
 - `padding: [top] [right= left] [bottom];`
 - `padding: [top=bottom] [right= left];`
 - `padding: [top=bottom=right= left];`

Dimensiuni, vizibilitate, afisare

- **height**: auto | [valoare] | [%] | inherit
- **width**: auto | [valoare] | [%] | inherit
- CSS 2: **min-height**, **max-height**, **min-width**, **max-width**: none(max) | [valoare] | [%] | inherit
- **visibility**: visible | hidden;
- **display**: none | inline | block;

Pozitionare

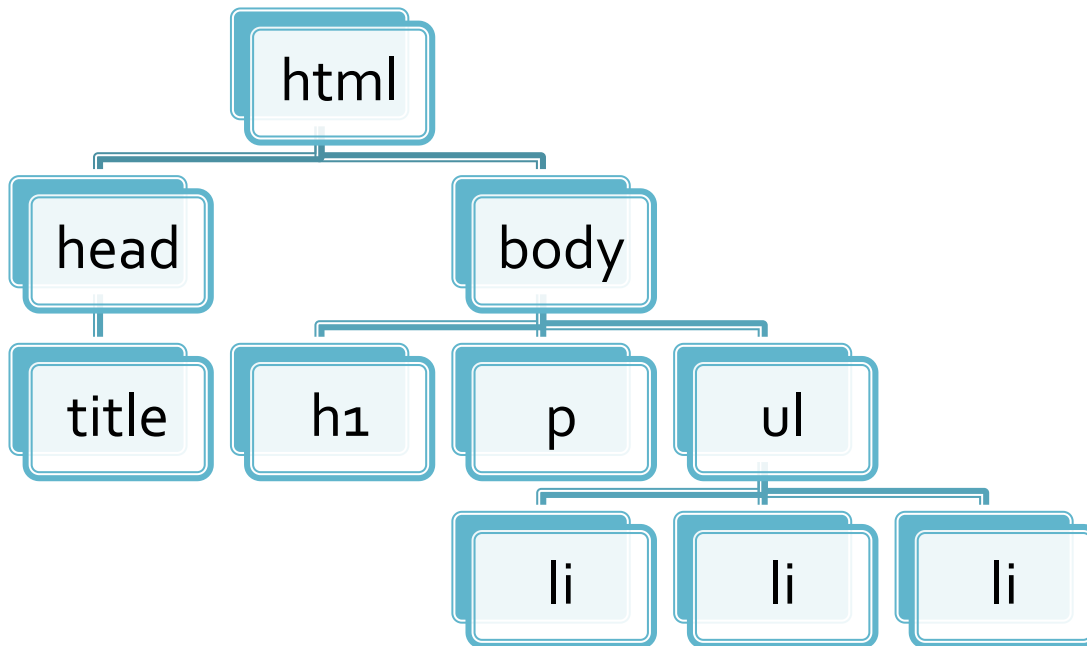
- **position**:absolute | fixed | relative | static | inherit;
- **top, right, bottom, left**:auto | [valoare] | [%] | inherit;
- **z-index**: [valoare intreaga]; -> mai mare = deasupra

Pozitionare

- Este **esentiala** in CSS
- Forma si structura vizuala a documentelor HTML este controlata foarte precis prin:
 - separarea continutului in zone dreptunghiulare, (uneori artificial create cu <div>)
 - pozitionarea foarte exacta a acestor zone, eventual unele in raport cu altele

DOM

- DOM – Document Object Model: structura de tip graf



```
<html>
  <head>
    <title>pagina mea</title>
  </head>
  <body>
    <h1>Compozitori:</h1>
    <p>
      <ul>
        <li> elvis costello
        <li> johannes brahms
        <li> georges brassens
      </ul>
    </p>
  </body>
</html>
```


Relatie cu Javascript in aplicatii

- Javascript poate accesa prin intermediul obiectului DOM atasat documentului HTML elementele din structura arbore DOM si modifica proprietatile corespunzatoare
- se deschide astfel calea spre aplicatii dinamice

CSS Zen Garden

CSS Zen Garden – HTML original

css Zen Garden

The Beauty of CSS Design

A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the

Download the sample [html file](#) and [css file](#)

The Road to Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless effort

The CSS Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with your eyes. Become one with the web.

So What is This About?

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and educate. When you load the style sheet into this very page. The code remains the same, the only thing that has changed is the

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated is by placing it in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hacks

CSS Zen Garden



Css Zen Garden

the beauty of css design.

A demonstration of what can be accomplished visually through CSS-based design. Select any style sheet from the list to load it into this page.

[\[X\] DOWNLOAD THE SAMPLE HTML FILE AND CSS FILE](#)



The Road To Enlightenment

Littering a dark and dreary road lay the past relics of browser-specific tags, incompatible DOMs, and broken CSS support.

Today, we must clear the mind of past practices. Web enlightenment has been achieved thanks to the tireless efforts of folk like the W3C, WaSP, and the major browser creators.

The css Zen Garden invites you to relax and meditate on the important lessons of the masters. Begin to see with clarity. Learn to use the (yet to be) time-honored techniques in new and invigorating fashion. Become one with the web.



So What is This About

There is clearly a need for CSS to be taken seriously by graphic artists. The Zen Garden aims to excite, inspire, and encourage participation. To begin, view some of the existing designs in the list. Clicking on any one will load the style sheet into this very page. The code remains the same, the only thing that has changed is the external .css file. Yes, really.

CSS allows complete and total control over the style of a hypertext document. The only way this can be illustrated in a way that gets people excited is by demonstrating what it can truly be, once the reins are placed in the hands of those able to create beauty from structure. To date, most examples of neat tricks and hacks have been

SELECT A DESIGN:

- ☐ [Under the Seal](#)
by Eric Stoltz
- ☐ [Make 'em Proud](#)
by Michael McGlough and
Scotty Reifsnnyder
- ☐ [Orchid Beauty](#)
by Kevin Addison
- ☐ [Oceanscape](#)
by Justin Gray
- ☐ [CSS Co., Ltd.](#)
by Benjamin Klemm
- ☐ [Sakura](#)
by Tatsuya Uchida
- ☐ [Kyoto Forest](#)
by John Polkowski
- ☐ [A Walk in the
Garden](#)
by Simon Van
Hauwermeiren

ARCHIVES:

[next designs »](#)

[View All Designs](#)

RESOURCES:

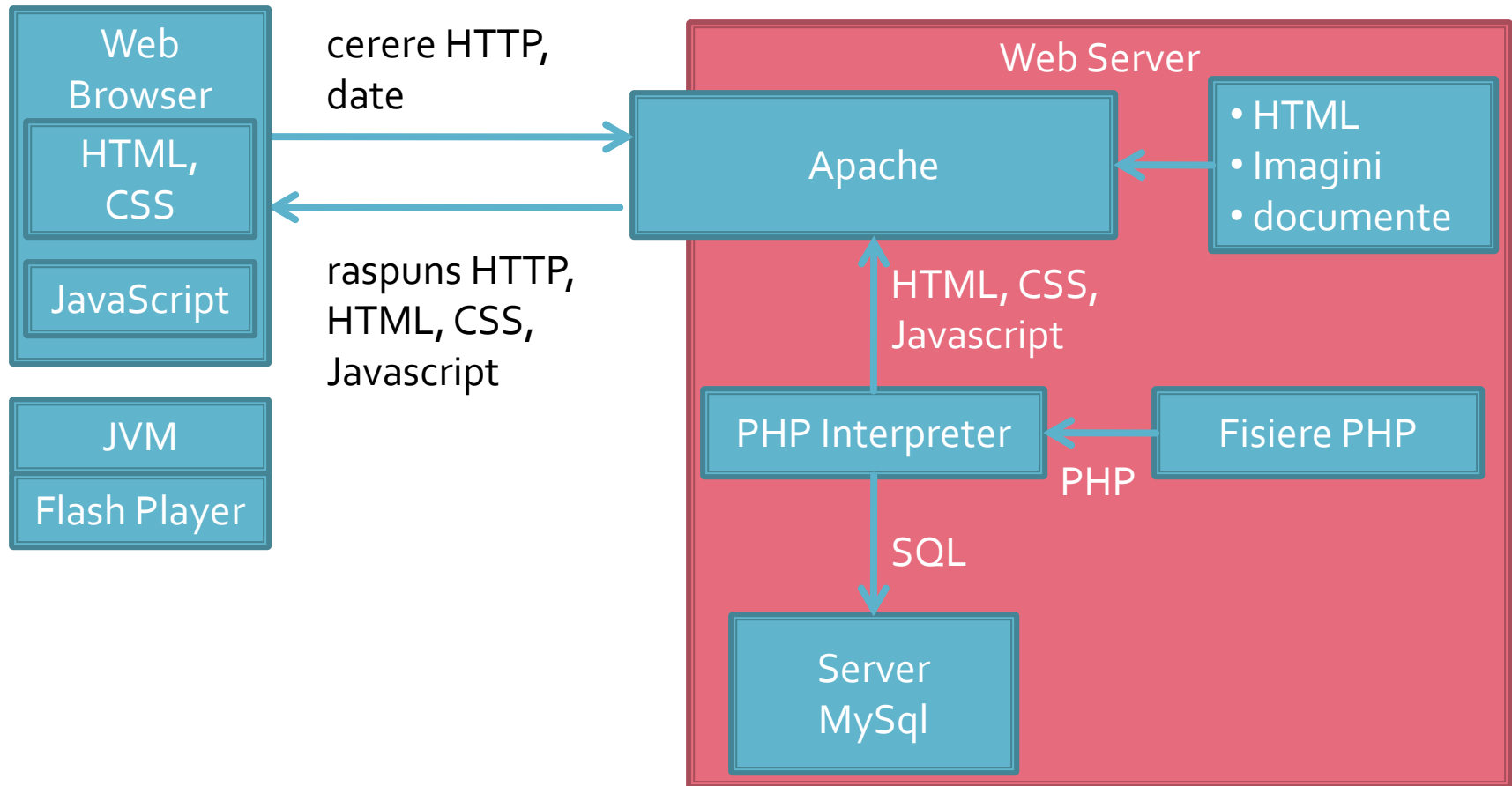
Hypertext PreProcessor

PHP

PHP

- Hypertext PreProcessor - acronim recursiv
 - initial – Personal Home Page / Form Interpreter
 - 1995 – 1.0
- versiune curenta: 8.3.3
 - 2024-02-15
- versiune curenta: 8.2.16
 - 2024-02-15
- limbaj de scripting de uz general,
- rulare pe server ([server-side scripting](#))
- open source

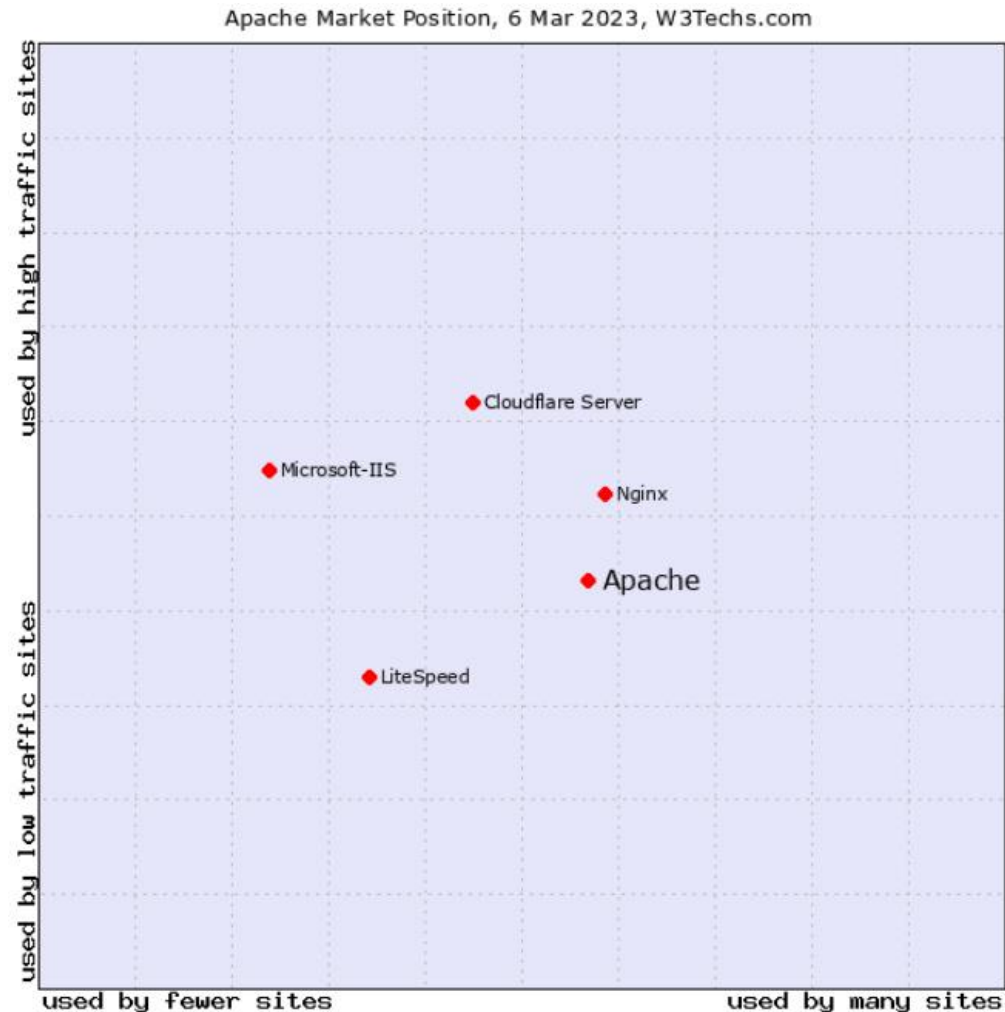
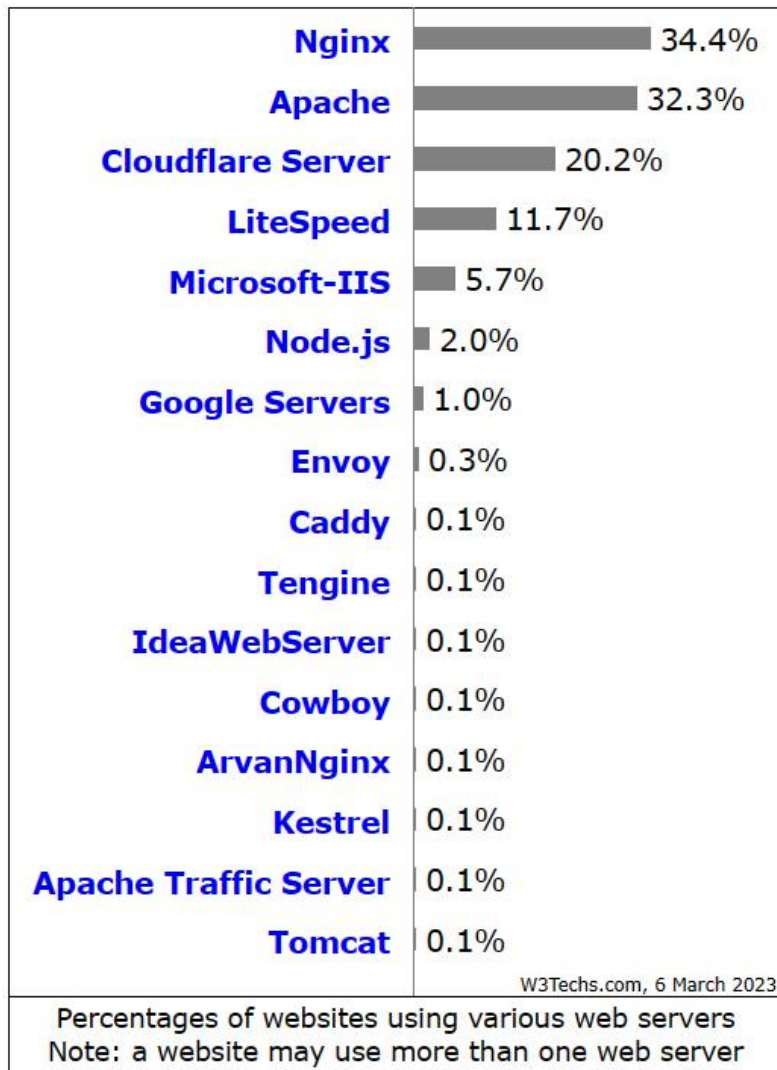
Client/Server Scripting



client
side
scripting

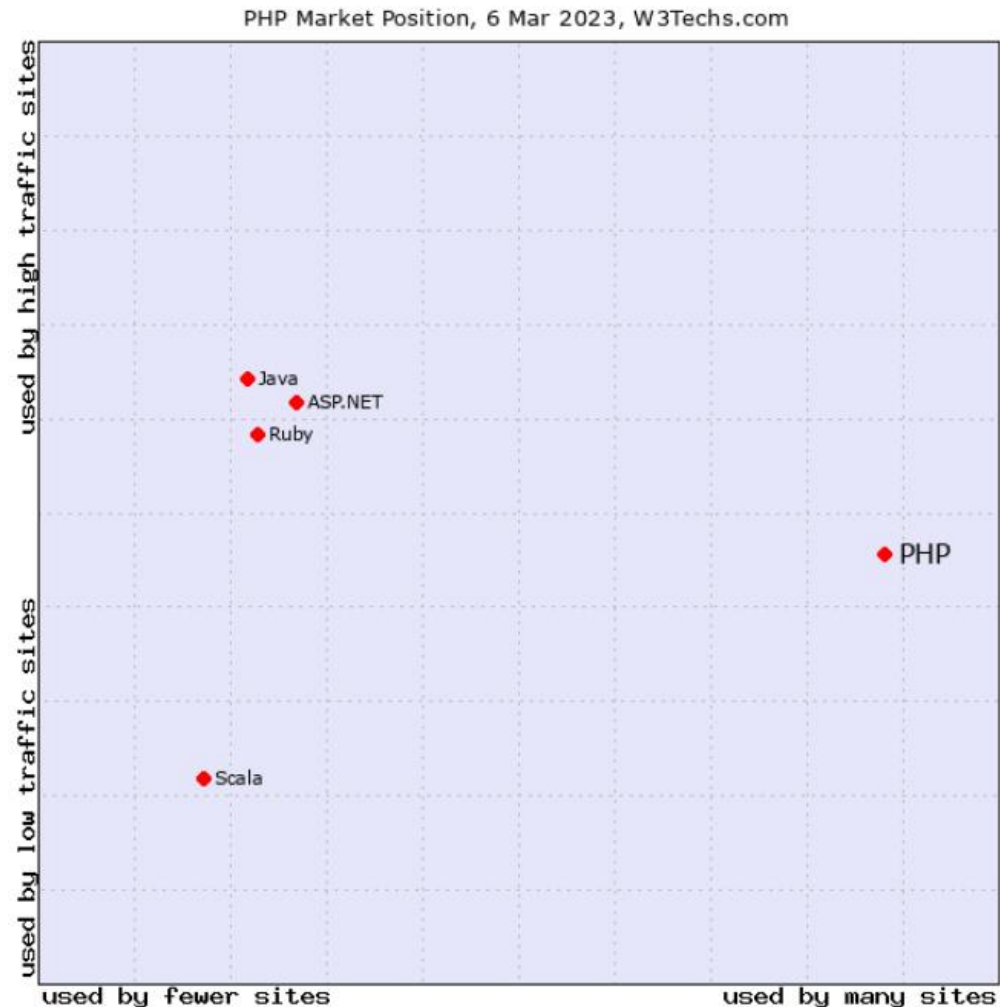
server
side
scripting

Tehnologia server-elor Web



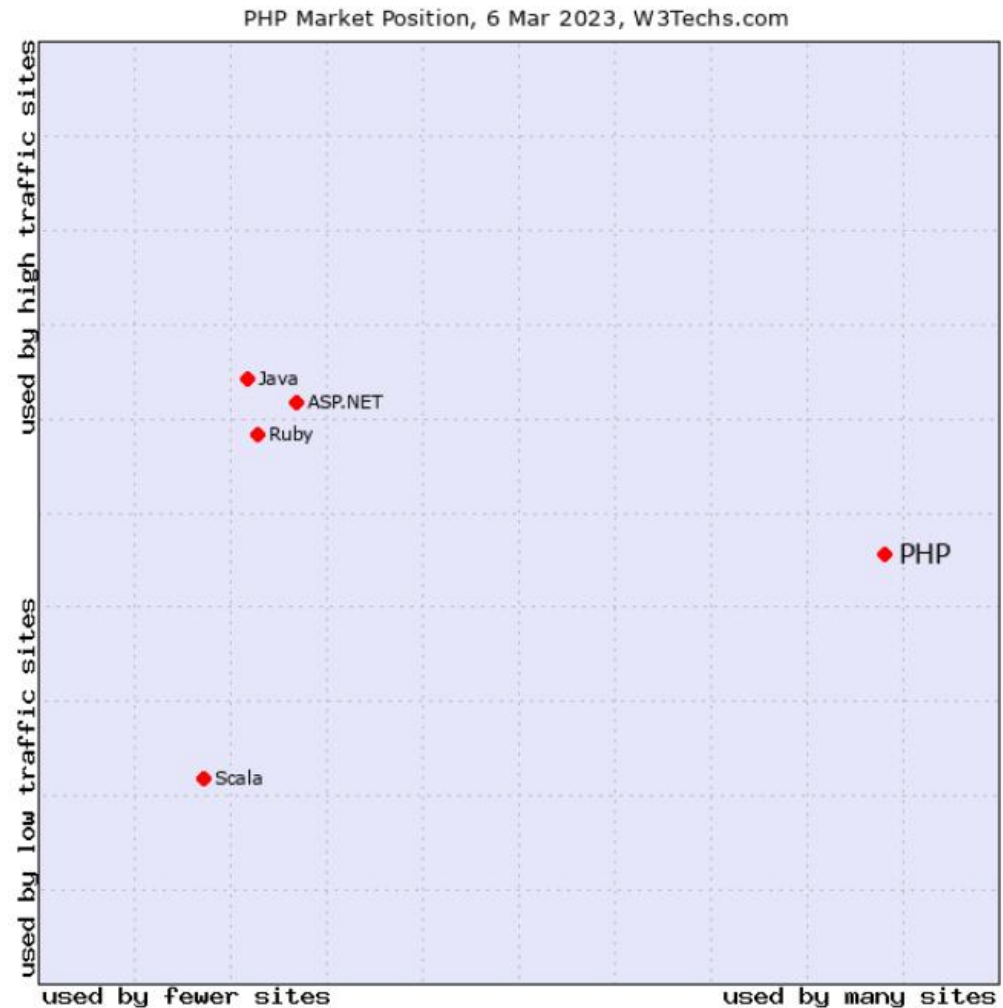
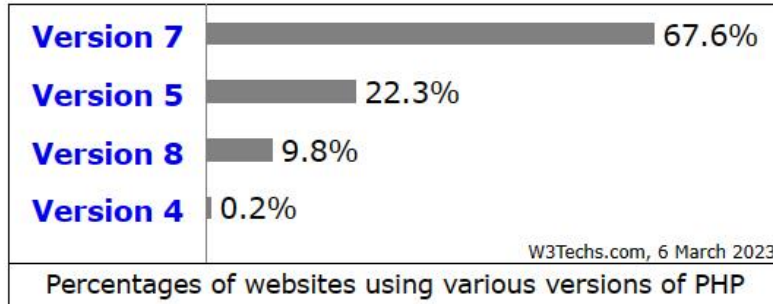
Tehnologia server-elor Web

- PHP – 77.50%
- ASP.NET – 7.20%
- Ruby – 5.20%
- Java – 4.70%
- Scala – 2.90%
- JavaScript – 2.00%
- ...
- Python – 1.50%



Tehnologia server-elor Web

■ PHP version



Server Side Scripting

	2010 1 Jan	2011 1 Jan	2012 1 Jan	2013 1 Jan	2014 1 Jan	2015 1 Jan
PHP	72.5%	75.3%	77.3%	78.7%	81.6%	80.6%
ASP.NET	24.4%	23.4%	21.7%	20.2%	18.2%	16.7%
Java	4.0%	3.8%	4.0%	4.1%	2.7%	2.8%
ColdFusion		1.3%	1.2%	1.1%	0.8%	0.7%
Perl		1.1%	1.0%	0.8%	0.6%	0.5%
Ruby	0.5%	0.5%	0.6%	0.5%	0.4%	0.9%
Python	0.3%	1.0%	1.3%	1.5%	1.7%	1.6%
JavaScript			<0.1%	<0.1%	0.1%	0.1%

Sites

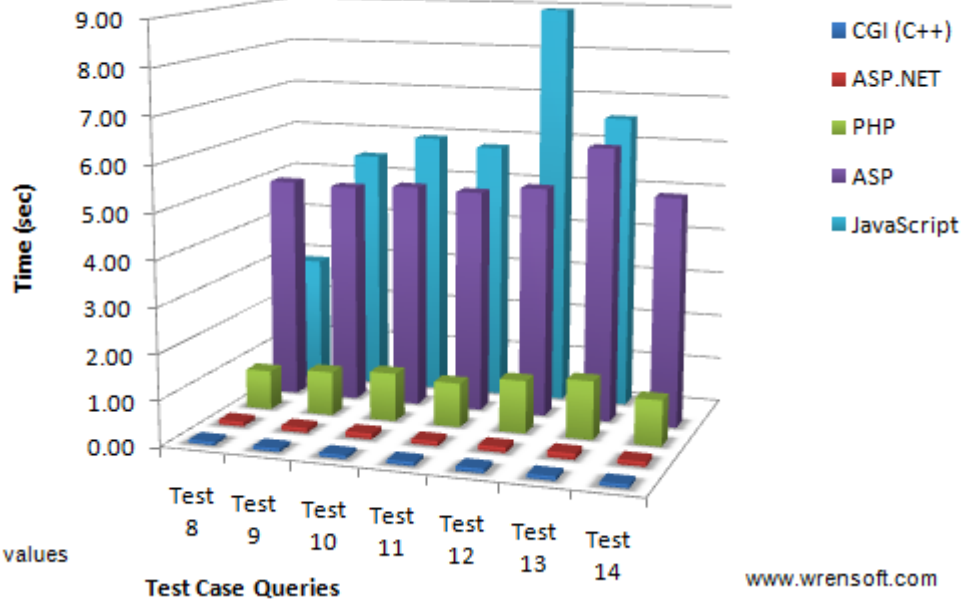
- Facebook.com
- Wikipedia.org
- Wordpress.org
- Pinterest.com
- Vimeo.com
- Wordpress.com
- Mozilla.org
- Europa.eu
- Tumblr.com
- Digicert.com

Avantaje PHP

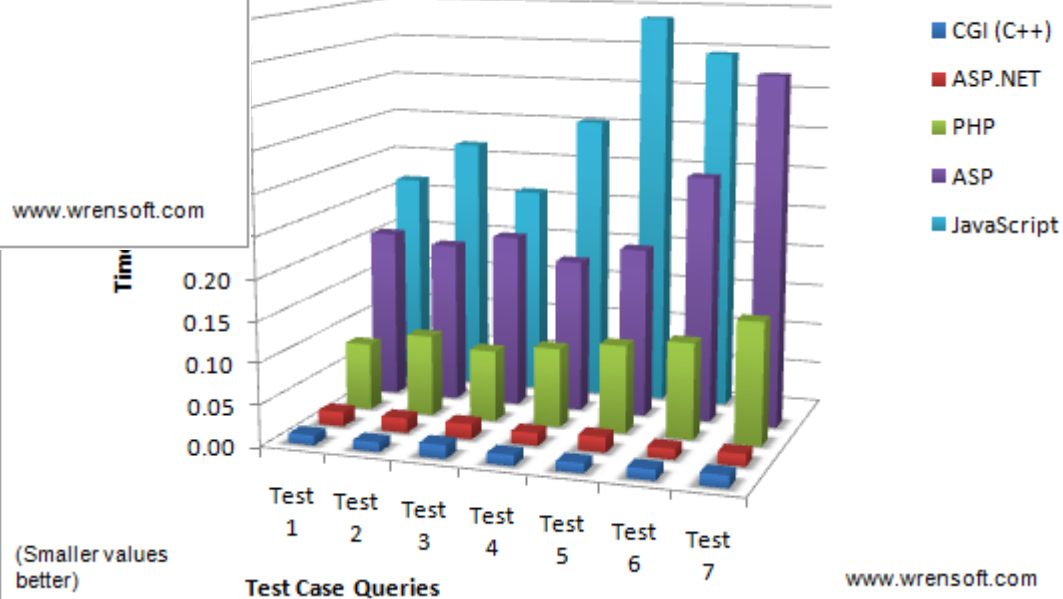
- Performanta ridicata
- Interfata cu multe sisteme de baze de date
- Costuri reduse
- Biblioteci incluse pentru majoritatea operatiunilor uzuale in aplicatii web
- Usurinta in invatare (C)
- Portabilitate
 - Disponibilitate a codului sursa
- Exemple disponibile in comunitate
- Suport disponibil

Performanta

Zoom search benchmarking
(Medium site - 60K pages, 2.5 million words)



Zoom search benchmarking
(Small site - 400 pages, 266K words)



PHP - Concepte

- limbaj interpretat – compilat “on the fly” de interpretorul PHP de pe server
- script-urile PHP contin sursele
 - exista posibilitatea pre-compilarii surselor pentru spor de viteza
 - Hip-Hop for PHP / Facebook
 - HipHop Virtual Machine / Facebook
- orientat spre aplicatii web dinamice (biblioteci)
- poate fi integrat in HTML – utilizarea tipica

Integrare

```
E:\Documents\Pagini Web\Safir dep\lista_lot.php - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
logfile.php lista_lot.php
539 <?php }?>
540
541 <div id="maincontent"><!-- InstanceBeginEditable name="Continut" -->
542 <p><span class="title">Adauga</span><br/>
543 <a href="control_lot.php?id=0">Lot nou</a></p>
544 <p class="title">Loturi active</p>
545 <?php
546 $query = "SELECT l.*, c.`nume_user` AS `user_creat`
547 FROM `lot` AS l
548 LEFT JOIN `users` AS c ON (l.`User`=c.`id_user`)
549 WHERE l.`Activ` <> 0 ORDER BY l.`ID_LOT` DESC";
550 $result = mysql_query($query);
551 $total=0;
552 if ($result && (mysql_num_rows($result) > 0))
553 {
554 $total=mysql_num_rows($result);
555 $row = mysql_fetch_assoc($result);
556 }
557 if ($total>0)
558 {?>
559 <table align="center">
560 <tr class="lista_titlu">
561 <td align="center">Nr.</td>
562 <td><?php echo $row['Numar'];?>&nbsp;</td>
563 <td><?php echo $row['Data'];?>&nbsp;</td>
564 <td><?php echo date("d/m/Y", strtotime($row['Data']));?>&nbsp;</td>
565 <td><?php echo $row['Pui'];?>&nbsp;</td>
566 <td><?php echo $row['Greutate'];?>&nbsp;</td>
567 <td><?php echo $row['Pui_Morti'];?>&nbsp;</td>
568 <td><?php echo $row['Greutate_Morti'];?>&nbsp;</td>
569 <td><?php echo $row['Ferma'];?>&nbsp;</td>
570 <td><?php echo $row['Documente'];?>&nbsp;</td>
571 <td><?php echo $row['Comenzi'];?>&nbsp;</td>
572 </tr>
573 <?php $index=1;
574 do { ?>
575 <tr class="<?php if ($index%2) echo "lista_impar"; else echo "lista_p
576 <td align="center"><?php echo $index; ?>&nbsp;</td>
577 <td><?php echo $row['Numar'];?>&nbsp;</td>
578 <td><?php echo date("d/m/Y", strtotime($row['Data']));?>&nbsp;</td>
579 <td><?php echo $row['Pui'];?>&nbsp;</td>
580 <td><?php echo $row['Greutate'];?>&nbsp;</td>
581 <td><?php echo $row['Pui_Morti'];?>&nbsp;</td>
582 <td><?php echo $row['Greutate_Morti'];?>&nbsp;</td>
583 <td><?php echo $row['Ferma'];?>&nbsp;</td>
584 <td><?php echo $row['Documente'];?>&nbsp;</td>
```


Separare cod PHP

- un fisier sursa PHP **este** un fisier HTML (in general) cu sectiuni de cod PHP **in interior**
- interpretorul PHP cauta sectiunile pe care trebuie sa le interpreteze si in interiorul lor proceseaza instructiunile ca fiind PHP
- ce se gaseste in **exteriorul** acestor sectiuni este trimis spre server-ul web **nemodificat**
- de cele mai multe ori rezultatul procesarii este un "fisier text" virtual continand cod **HTML** (strict) (+CSS/Javascript) transmis(e) server-ului web pentru distribuire

Separare cod PHP

- `<?php ... ?>`
 - stil XML – implicit, disponibil intotdeauna, recomandat
- `<? ... ?>`
 - scurt, este de obicei dezactivat
 - lipsa in PHP 7,8
- `<script language="php"> ... </script>`
 - stil script, disponibil (?)
- `<% ... %>`
 - stil ASP, de obicei dezactivat
 - lipsa in PHP 7,8

Variante de integrare

- echo afiseaza un text la "iesire" (echivalent cu puts() din C)
- poate realiza procesarea datelor
- in exemplu se trimite spre iesire un sir static (echivalent cu puts() din C)
- "iesire" in marea majoritate a cazurilor reprezinta datele trimise clientului de server-ul web
- "iesire" poate fi considerata de obicei:
 - documentul curent
 - pozitia curenta (a scriptului in fisierul HTML curent)

Variante de integrare

- Toate variantele ofera aceeasi sursa HTML pentru browser
- E **recomandata** cea care lasa structura HTML nemodificata si doar datele dinamice sunt rezultatul procesarii
- Codul HTML + PHP e interpretat mult mai elegant in editoarele WYSIWYG

```
<h2>Rezultate comanda</h2>  
<?php echo '<p>Comanda receptionata</p>';?>
```

```
<h2>Rezultate comanda</h2>  
<p><?php echo 'Comanda receptionata';?></p>
```

```
<?php echo '<h1>Magazin online XXX SRL</h1>';?>  
<?php echo '<h2>Rezultate comanda</h2>';?>  
<?php echo '<p>Comanda receptionata</p>';?>
```

```
<?php  
echo '<h1>Magazin online XXX SRL</h1>';  
echo '<h2>Rezultate comanda</h2>';  
echo '<p>Comanda receptionata</p>';  
?>
```

Variante de integrare

```
test.php x
1 <h2>Rezultate comanda</h2>
2 <p><?php echo 'Comanda receptionata';?></p>
3
4 <?php
5 echo '<h1>Magazin online XXX SRL</h1>';
6 echo '<h2>Rezultate comanda</h2>';
7 echo '<p>Comanda receptionata</p>';
8 ?>
9
```

PHP – instructiuni

- instructiunile PHP trebuie sa se termine cu ;
 - exceptie: se poate omite ; la sfarsitul blocului de cod php: ... **echo 'ceva' ?>**
- instructiunile pot sa fie scrise pe acelasi rand (fara trecerea la linia noua)
 - **echo 'ceva1'; echo 'ceva2'; ...**
 - nu este in general recomandat

PHP – comentarii

- comentariile in PHP respecta regulile C si Perl
- la sfarsit de linie:
 - `// echo ceva; //comentariu`
 - `# echo ceva; #comentariu`
- comentariu bloc
 - `/* ... */`
 - `/* un comentariu
pe mai multe
linii */`

PHP – constante

- Ca orice limbaj de programare PHP se bazeaza pe utilizarea
 - constante
 - variabile
 - functii
- Definirea constantelor:
 - `define('PRETCARTE', 100);`
 - "case sensitive"
 - prin conventie, numai cu litere mari
 - `echo PRETCARTE; // 100`

PHP – variabile

- variabila – semnul **\$** urmat de un nume
- numele e “case sensitive”
- o greseala frecventa e uitarea semnului \$
 - PHP Notice: Use of undefined constant an – assumed \$an (**sau 'an'**) in D:\\Server\\
- Tipuri de date
 - scalar
 - compus
 - special

PHP – tipuri de date

- scalar
 - boolean
 - integer
 - float (double)
 - **string**
- compus
 - array
 - object
- special
 - resource
 - NULL

PHP – tipuri de date

- **declararea** variabilelor **nu** e necesara decat cand se declara un domeniu de definitie (variabile globale)
 - `global $a, $b;`
`$c=$a+$b;`
- eliberarea memoriei nu este necesara, se face automat la terminarea executiei

PHP – tipuri de date

```
$var = expresie
```

- Controlul variabilelor se face automat, “on the fly”
 - Daca \$var nu era definita anterior, in urma atribuirii se defineste de tipul dat de rezultatul expresiei
 - Daca \$var era definita, de un anumit tip (oarecare), in urma atribuirii devine de tipul dat de rezultatul expresiei
 - La finalizarea executiei script-ului se elimina variabila din memorie (automat)
 - Numele de variabile sunt "case-sensitive" (\$age si \$AGE sunt doua variabile diferite)

PHP – tipuri de date

- tipul de date este in totalitate dependent de ceea ce se stocheaza
- PHP reactualizeaza tipul pentru a putea primi ceea ce se stocheaza

```
<?php
echo $variabila ; // tip Null, neinitializat – valoare NULL (doar)
$variabila = "0"; // $variabila tip string (ASCII 48)
$variabila += 2; // $variabila tip integer (2)
$variabila = $variabila + 1.3; // $variabila tip float (3.3)
$variabila = 5 + "10 obiecte"; // $variabila tip integer (15)
$var2=5; // $var2 tip integer (5)
$variabila=$var2."10 obiecte"; // $variabila tip string "510 obiecte"
?>
```

PHP – operatori

- In general similari celor din C/C++
- Operatori
 - Aritmetici
 - Atribuire
 - Bit
 - Comparare
 - Incrementare/Decrementare
 - Logici
 - **Sir**

PHP – operatori

- Aritmetici
 - $-$a$ – Negare
 - $$a + b – Adunare
 - $$a - b – Scadere
 - $$a * b – Inmultire
 - $$a / b Impartire
 - $$a \% b Modulo (rest)
- Sir
 - **$$a.b – Concatenare sir a si sir b**

PHP – operatori

■ Atribuire

- `$a=$b`
- `$a+=$b` ($a=a+b$)
- `$a-=$b` ($a=a-b$)
- `$a/=$b` ($a=a/b$)
- `$a*=$b` ($a=a*b$)
- `$a%=$b` ($a=a\%b$)
- `$a.=$b` ($a=a$ concatenat b - siruri)

PHP – operatori

- Operatori la nivel de bit
 - similari celor din C
 - `~, &, |, ^, <<, >>`
- Operatori logici
 - ofera rezultat boolean true/false
 - similari celor din C
 - `&&, ||, !`
 - suplimentar
 - `and, or, xor` – echivalenti dar de prioritate mai mica
 - `$a=55/0 or die('impartire prin 0');`

PHP – operatori

- Operatori de comparare
 - ofera rezultat boolean true/false
 - similari celor din C
 - == , != , > , < , <> , >= , <=
 - suplimentar
 - === identic, valoare egala **SI** de acelasi tip
 - !== “neidentic”, valoare diferita **SAU** de tipuri diferite

Precedenta operatorilor

non-associative	clone new	clone and new
left	[array()
non-associative	++ --	increment/decrement
right	~ - (int) (float) (string) (array) (object) (bool) @	types
non-associative	instanceof	types
right	!	logical
left	* / %	arithmetic
left	+ - .	arithmetic and string
left	<< >>	bitwise
non-associative	< <= > >= <>	comparison
non-associative	== != === !==	comparison
left	&	bitwise and references
left	^	bitwise
left		bitwise
left	&&	logical
left		logical
left	? :	ternary
right	= += -= *= /= .= %= &= = ^= <<= >>=	assignment
left	and	logical
left	xor	logical
left	or	logical
left	,	many uses

Precedenta operatorilor PHP7

non-associative	clone new	clone and new
left	[array()
non-associative	++ --	increment/decrement
right	~ - (int) (float) (string) (array) (object) (bool) @	types
non-associative	instanceof	types
right	!	logical
left	* / %	arithmetic
left	+ - .	arithmetic and string
left	<< >>	bitwise
non-associative	< <= > >= <>	comparison
non-associative	== != === !== <=>	comparison
left	&	bitwise and references
left	^	bitwise
left		bitwise
left	&&	logical
left		logical
right	??	comparison
left	? :	ternary
right	= += -= *= /= .= %= &= = ^= <<= >>=	assignment
left	and	logical
left	xor	logical
left	or	logical
left	,	many uses

Precedenta operatorilor PHP8

non-associative	clone new	clone and new
right	**	arithmetic
non-associative	++ --	increment/decrement
right	~ - (int) (float) (string) (array) (object) (bool) @	types
left	instanceof	types
right	!	logical
left	* / %	arithmetic
left	+ -	arithmetic
left	<< >>	bitwise
left	.	string
non-associative	< <= > >= <>	comparison
non-associative	== != === !== =<=>	comparison
left	&	bitwise and references
left	^	bitwise
left		bitwise
left	&&	logical
left		logical
right	??	comparison
non-associative	? :	ternary
right	= += -= *= /= .= %= &= = ^= <<= >>=	assignment
left	and	logical
left	xor	logical
left	or	logical
left	,	many uses

Exemplu de separare cod php

Raspuns dinamic

- `<form action="rezultat.php" method="post">`

```
<html>
<head>
<title>Magazin online XXX SRL</title>
</head>
<body>
<h1>Magazin online XXX SRL</h1>
<h2>Rezultate comanda</h2>
<p><?php echo 'Comanda receptionata';?></p>
</body>
</html>
```

Magazin online XXX SRL

Rezultate comanda

Comanda receptionata

```
<
<
<
<
</head>
<body>
<h1>Magazin online XXX SRL</h1>
<h2>Rezultate comanda</h2>
<p>Comanda receptionata</p>
</body>
</html>
```

PHP – Functii

- conceptual similare celor din C/C++
- functiile nu trebuie declarate inainte de a fi folosite
- numele functiilor este “case-insensitive”
- un mare numar de functii cu utilitate directa in aplicatiile web exista in bibliotecile PHP
- unele biblioteci trebuie activate in momentul configurarii PHP
 - `extension=php_gd2.dll` (linia 639) // pentru functii de procesare grafica de exemplu
 - `extension=php_mysql.dll` (linia 651) // pentru functii de acces la baze de date MySql
 - `extension=php_mysqli.dll` (linia 652) // pentru functii de acces la baze de date MySql (**obligatoriu** de la PHP 5.6)

Utilizarea functiilor PHP

- `<form action="rezultat.php" method="post">`

`<p>Comanda receptionata la data:`

`<?php echo date('d/m/Y')." ora ".date('H:i');?></p>`

Magazin online XXX SRL

Rezultate comanda

Comanda receptionata la data: 10/03/2010 ora 13:36

`<body>`

`<h1>Magazin online XXX SRL</h1>`

`<h2>Rezultate comanda</h2>`

`<p>Comanda receptionata la data:`

`10/03/2010 ora 13:36</p>`

`</body>`

Elemente de control

- majoritatea notiunilor si sintaxei sunt similare celor din C/C++
- instructiune compusa: separata de acolade {...}
- if / else / elseif – executie conditionata

```
<?php
if ($a > $b) {
    echo "a mai mare ca b";
} elseif ($a == $b) {
    echo "a egal cu b";
} else {
    echo "a mai mic ca b";
}
?>
```

Elemente de control

- while
- do-while
- for
- switch
- return
- break
- goto
- Similare cu echivalentele C/C++

```
$i = 1;  
while ($i <= 10) {  
    echo $i++;  
}
```

```
$i = 10;  
do {  
    echo $i--;  
} while ($i > 0);
```

```
for ($i = 1; $i <= 10; $i++) {  
    echo $i;  
}
```

```
switch ($i) {  
    case 0:  
        echo "i este 0";  
        break;  
    case 1:  
        echo "i este 1";  
        break;  
    default:  
        echo "i nici 1 nici 0";  
        break;  
}
```

Elemente de control

- `include()`
- `require()`
- `include_once()`
- `require_once()`

- pentru inserarea **SI** evaluarea fisierului folosit ca parametru
- folosite pentru a nu multiplica sectiunile de cod comune
- `require` opreste executia script-ului curent daca fisierul parametru nu este gasit
- `..._once()` verifica daca respectivul fisier a mai fost introdus si **nu** il mai introduce inca o data

Variabile globale

Variabile globale

- Variabilele globale (predefinite)
 - accesibile script-urilor PHP prin conlucrarea cu server-ul
 - Exemple:
 - `$_SERVER` — Server and execution environment information
 - `$_GET` — HTTP GET variables
 - `$_POST` — HTTP POST variables
 - `$_FILES` — HTTP File Upload variables
 - `$_REQUEST` — HTTP Request variables
 - `$_SESSION` — Session variables
 - `$_ENV` — Environment variables
 - `$_COOKIE` — HTTP Cookies

Interactiunea cu utilizatorul

- Datele introduse de utilizator in forme se regasesc (in functie de metoda aleasa pentru forma) in una din variabilele:
 - `$_POST` – method="post"
 - `$_GET` – method="get"
 - `$_REQUEST` – ambele metode
- variabilele sunt **tablouri** cu **cheia** data de atributul **name** din forma HTML
 - `<input type="text" name="carti_cant" size="3" maxlength="3" />`
 - `$_POST['carti_cant']` contine valoarea introdusa de utilizator

Laborator 2

HTML

- se creaza o pagina care sa arate ca in schita alaturata
- forma paginilor:
 - tabel
- Se experimenteaza diversele elemente de **interactiune cu utilizatorul** (**<form> + ...**)

culoare	IMAGINE	culoare
	Continut (cu alta culoare fundal)	

Suplimentar

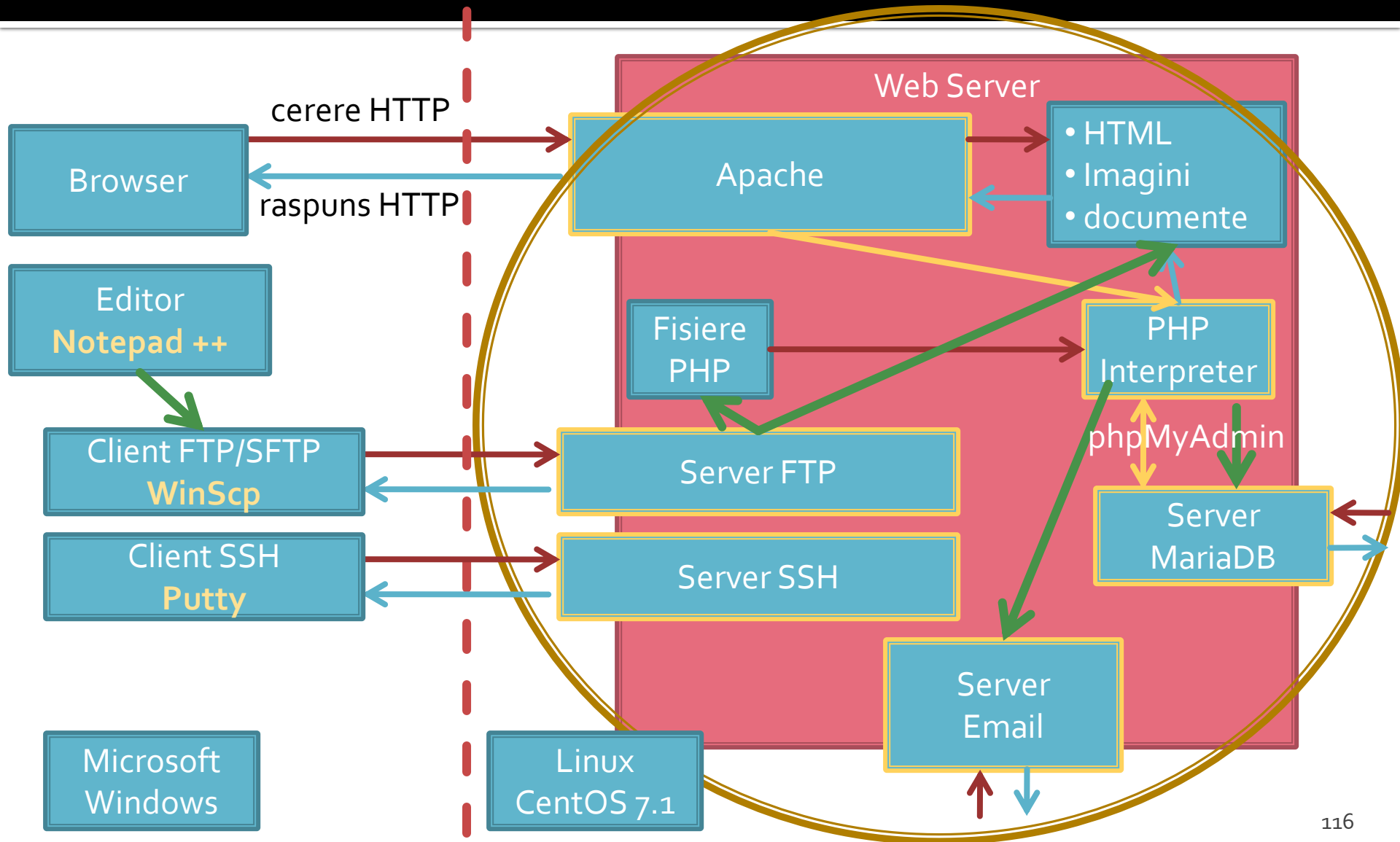
- forma din schita alaturata
- forma paginilor:
 - tabel controlat prin CSS

culoare	IMAGINE	culoare
	Continut (cu alta culoare fundal)	
	Copyright (cu alta culoare fundal)	

Exemplu

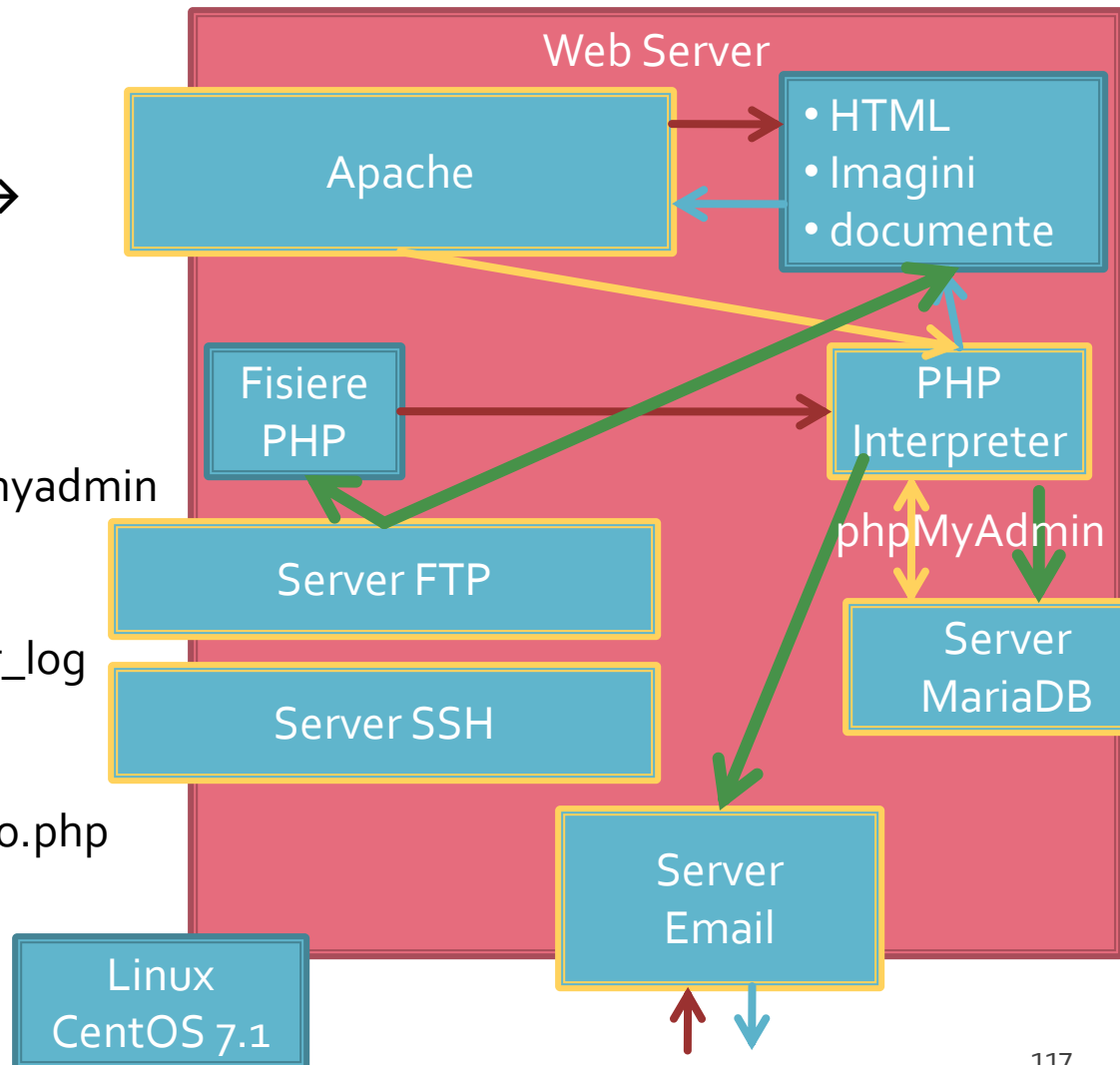
Magazin	Firma X SRL
Continut	

Utilizare LAMP



Utilizare LAMP

1. login → root:masterc
2. ifconfig → 192.168.30.5
3. putty.exe → 192.168.30.5 → SSH → root:masterc (remote login)
4. [alte comenzi linux dorite]
5. FTP → Winscp → SFTP → student:masterc@192.168.30.5
6. MySql → http://192.168.30.5/phpmyadmin → root:masterc
7. Apache Error Log →
 - 7a. putty → nano /var/log/httpd/error_log
 - 7b. http://192.168.30.5/logfile.php (nonstandard)
8. PHP info → http://192.168.30.5/info.php



Server referinta LAMP 2024

- 3 variante acceptate
 - CentOS 7.1
 - Ubuntu 20.04
 - Debian 12.5

Server referinta LAMP

- Centos 7.1
 - PHP 5.4.16
 - MariaDB 5.5.44 / root:masterrc
 - Apache 2.4.6
 - PhpMyAdmin/4.4.15
 - **root**/student:masterrc
 - Python 2.7.5

Server referinta LAMP

- Ubuntu 20.04
 - PHP 7.4.3
 - MariaDB 10.3.31 / root:masteretti
 - Apache 2.4.41
 - **paw**/student:masteretti
 - necesar suplimentar pentru **acces FTP user paw**:
 - `sudo usermod -a -G upload paw`
 - `sudo chmod -R 775 /var/www`
 - Python 3.8.10

Server referinta LAMP

- Debian 12.5
 - PHP 8.2.7
 - MariaDB 10.11.6 / root:masteretti
 - Apache 2.4.57
 - PhpMyAdmin/5.2.1 deb
 - **root**/paw/student:masteretti
 - Python 3.11.2

Server referinta

- rf-opto.etti.tuiasi.ro > Master > Web Design

Project/Design

~~[Teme proiect 2020 \(pdf, 874.95 KB, ro, 🇷🇴\)](#)~~

[Sevrer CentOS pentru VMWare Player \(cloud\)](#) (link, 0 Bytes, ro, 🇷🇴)

~~[Realizare Server CentOS \(pdf, 1.4 MB, en, 🇸🇪\)](#)~~

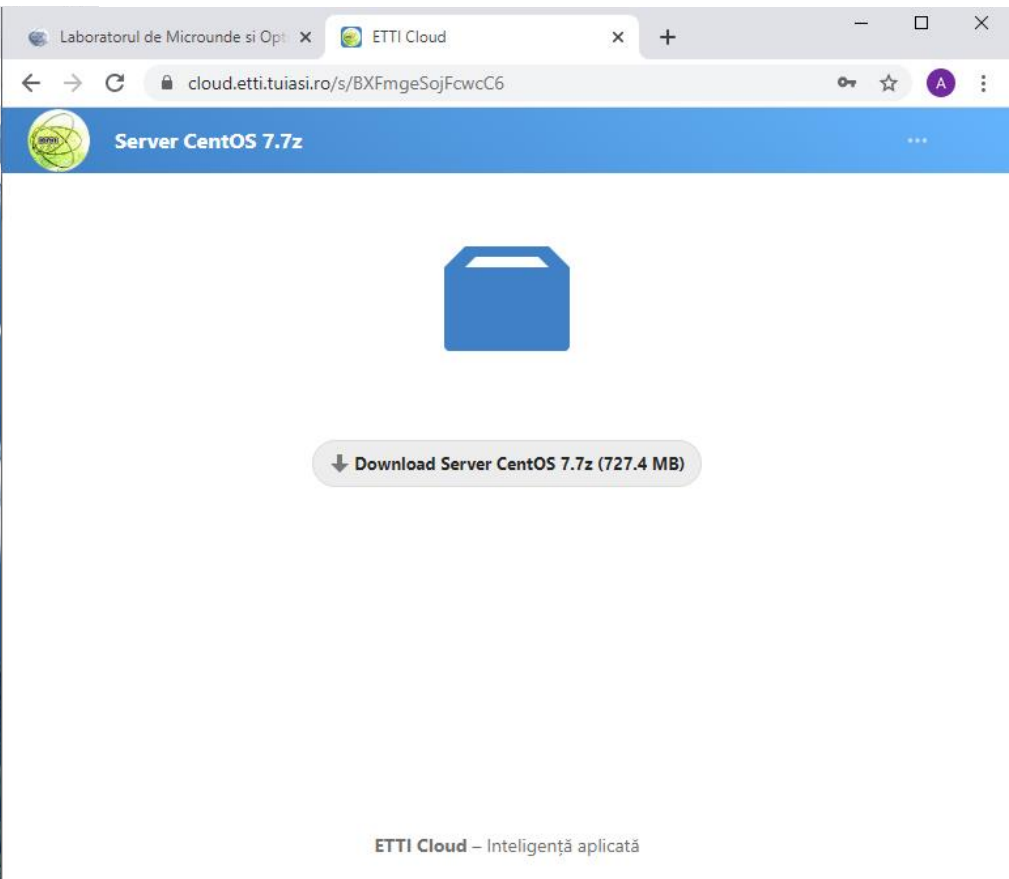
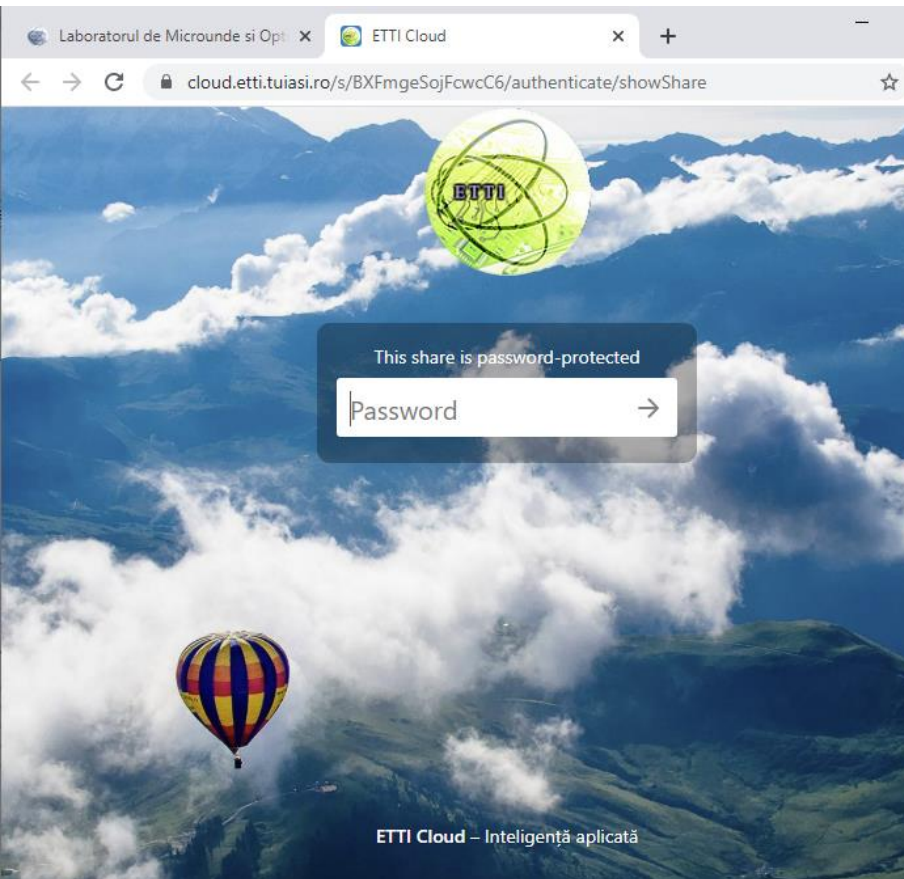
Examen

[Simulare Examen \(video\)](#) (mp4, 65.12 MB, ro, 🇷🇴)

[Manual examen on-line](#) (pdf, 2.65 MB, ro, 🇷🇴)

Server referinta

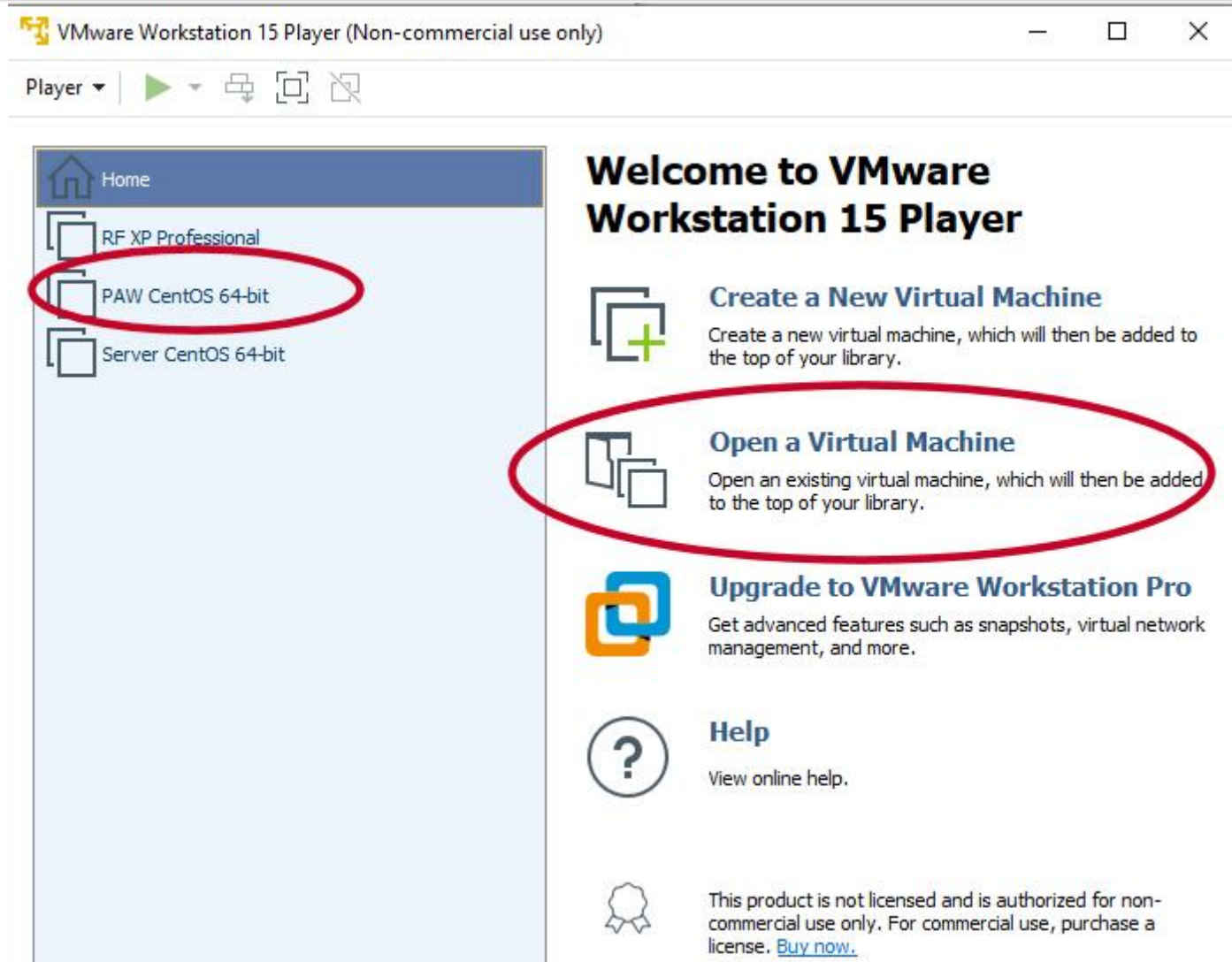
■ Cloud ETTI: RF-opto3#



Server referinta

- Masina virtuala
- VMware Workstation Player
 - Gratuit (non-comercial)
 - <https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html>

Server referinta



Aplicatii suport

- WinSCP (client FTP, gratuit)
 - <https://winscp.net/eng/download.php>
- Notepad ++ (editor, avansat, gratuit)
 - <https://notepad-plus-plus.org/downloads/>
- Putty (remote access)
 - <https://www.putty.org/>
- MySQL Workbench (gratuit, cont Oracle)
 - <https://www.mysql.com/products/workbench/>
- Visual Studio Code (gratuit, Microsoft)
 - <https://code.visualstudio.com/download>

Adresa IP

- login, ifconfig
- Ctrl + Alt + mouse

PAW CentOS 64-bit - VMware Workstation 15 Player (Non-commercial use only)

Player ▾ || ▾ ⏏ [] [X]

```
CentOS Linux 7 (Core)
Kernel 3.10.0-229.20.1.el7.x86_64 on an x86_64

tmpaw login: root
Password:
Last login: Wed Jun 17 05:35:16 from 192.168.0.106
[root@tmpaw ~]# ifconfig
```

PAW CentOS 64-bit - VMware Workstation 15 Player (Non-commercial use only)

Player ▾ || ▾ ⏏ [] [X]

```
CentOS Linux 7 (Core)
Kernel 3.10.0-229.20.1.el7.x86_64 on an x86_64

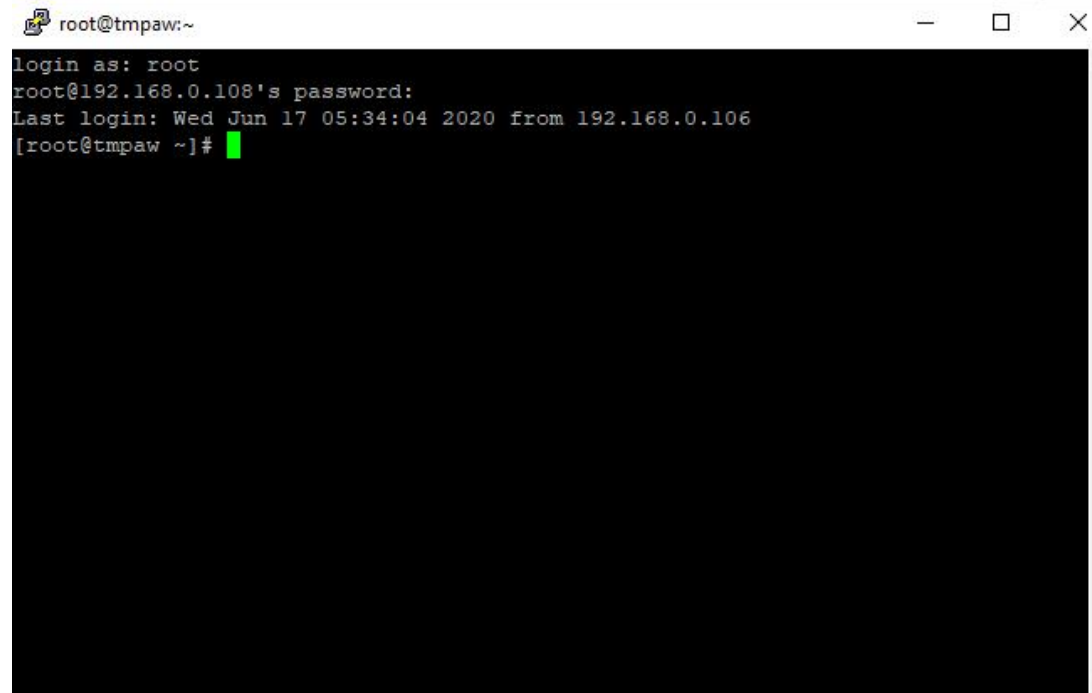
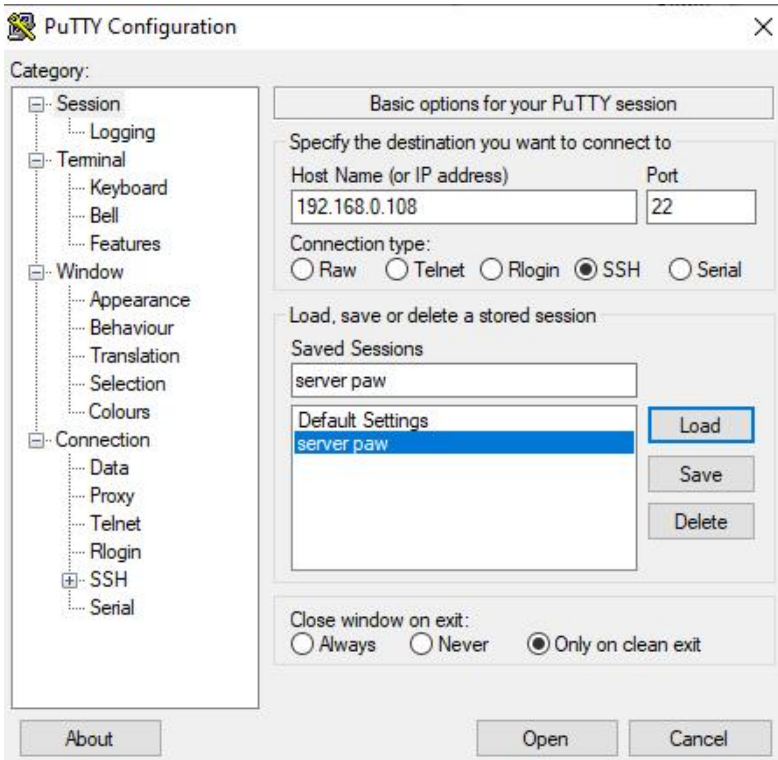
tmpaw login: root
Password:
Last login: Wed Jun 17 05:35:16 from 192.168.0.106
[root@tmpaw ~]# ifconfig
eno16777736: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.0.108 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::250:56ff:fe3e:1693 prefixlen 64 scopeid 0x20<link>
    ether 08:00:56:3e:16:93 txqueuelen 1000 (Ethernet)
    RX packets 104 bytes 12814 (12.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 99 bytes 11847 (11.5 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 0 (Local Loopback)
    RX packets 16 bytes 1774 (1.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 16 bytes 1774 (1.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[root@tmpaw ~]# _
```

Putty

- putty.exe
- evitare captura mouse, copy/paste etc.



WinSCP

- client FTP
- upload fisiere

Session

File protocol:
SFTP

Host name: 192.168.0.108 Port number: 22

User name: student Password:

Edit Advanced...

Login Close Help

html - student@192.168.0.108 - WinSCP

File Commands Mark Session View Help

Address /var/www/html

Find Files Download Edit Properties New Synchronize

Transfer Settings Default

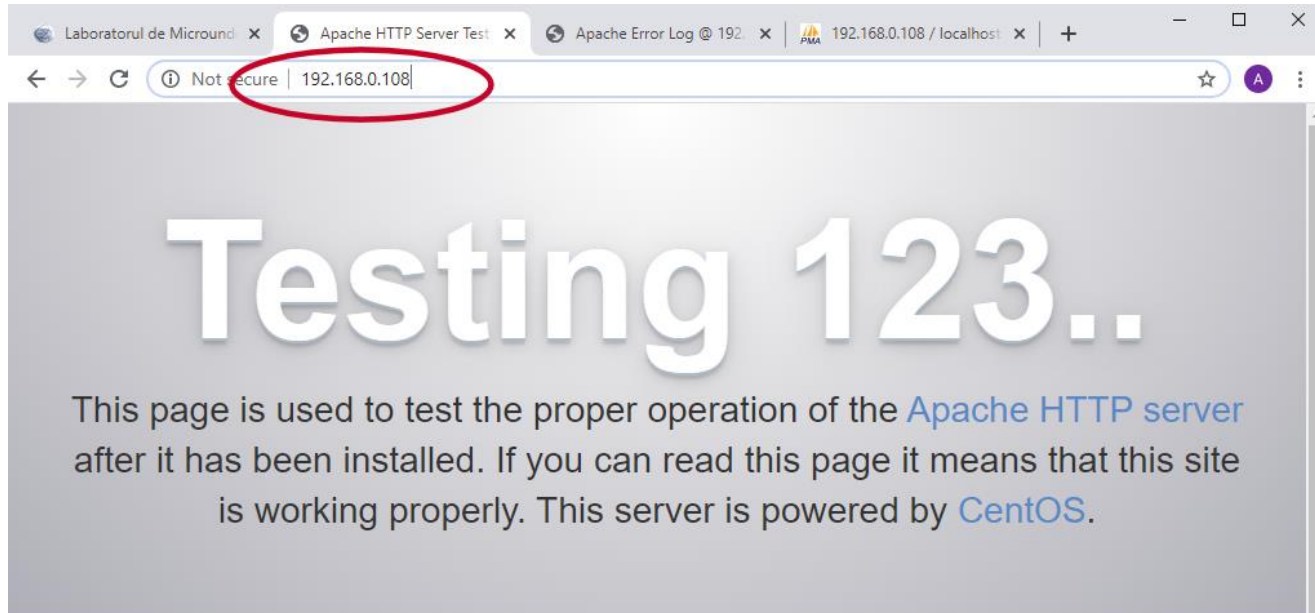
student@192.168.0.108 x New Session

Name	Size	Changed	Rights
ap.log	1 KB	2/29/2016 11:28:50 AM	rw-rw-r
info.php	1 KB	9/30/2009 3:23:00 PM	rw-rw-r
logfile.php	4 KB	12/6/2015 12:05:08 PM	rw-rw-r
test.php	2 KB	2/29/2016 12:04:12 PM	rw-rw-r

0 B of 5.09 KB in 0 of 4

SFTP-3 129 1, 21:06:30

Browser



Just visiting?

The website you just visited is either experiencing problems or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting `www.example.com`, you should send e-mail to `"webmaster@example.com"`.

Are you the Administrator?

You should add your website content to the directory `/var/www/html/`.

To prevent this page from ever being used, follow the instructions in the file `/etc/httpd/conf.d/welcome.conf`.

Promoting Apache and CentOS

You are free to use the images below on Apache and CentOS Linux powered HTTP servers. Thanks for using Apache and CentOS!



Server MySQL/MariaDB

The screenshot displays the phpMyAdmin web interface. The browser's address bar shows the URL `192.168.0.108/phpmyadmin/...`, with the domain part circled in red. The interface includes a left sidebar with a database tree, a top navigation bar with tabs like 'Databases', 'SQL', and 'Status', and a main content area with sections for 'General Settings', 'Appearance Settings', 'Database server', 'Web server', and 'phpMyAdmin'.

General Settings

- Change password
- Server connection collation: `utf8mb4_unicode_ci`

Appearance Settings

- Language: `English`
- Theme: `pmahomme`
- Font size: `82%`
- More settings

Database server

- Server: Localhost via UNIX socket
- Server type: MariaDB
- Server version: 5.5.44-MariaDB - MariaDB Server
- Protocol version: 10
- User: `root@localhost`
- Server charset: UTF-8 Unicode (utf8)

Web server

- Apache/2.4.6 (CentOS) OpenSSL/1.0.1e-fips mod_fcgid/2.3.9 PHP/5.4.16 mod_python/3.5.0- Python/2.7.5
- Database client version: libmysql - 5.5.44-MariaDB
- PHP extension: `mysql`
- PHP version: 5.4.16

phpMyAdmin

- Version information: 4.4.15.1
- Documentation
- Wiki
- Official Homepage
- Contribute
- Get support
- List of changes

Contact

- Laboratorul de microunde si optoelectronica
- <http://rf-opto.etti.tuiasi.ro>
- rdamian@etti.tuiasi.ro