

Laborator 3 (in loc de Curs 4)

2019/2020

Programarea aplicațiilor web

Introducere

PAW 2019-2020

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

Nota 2019-2020

- An V
 - 33% E
 - 66% Aplicatii
 - 33% L (0%)
 - 33% P (66%)

Documentatie

- <http://rf-opto.etti.tuiasi.ro> + WWW



The screenshot displays the website for RF-OPTO at ETTI. The header includes the ETTI logo, the text 'RF-OPTO', and the University of Technical Sciences 'Gheorghe Asachi' Iasi logo. A navigation menu contains 'Main', 'Courses', 'Master', 'Staff', 'Research', and 'Students'. Below the menu, there are sub-links for 'Radiocommunication Systems', 'Microwave IC', and 'Web Design'. The main content area features the course title 'Modern Web Applications Design' and the following details:

Course: TMPAW (2014-2015)
Course Coordinator: Assist.P. Dr. Radu-Florin Damian
Code: DS507RC
Discipline Type: DOMS; Alternative, Synthesis
Credits: 7
Enrollment Year: 5, Sem. 10

Activities
Course: Instructor: Assist.P. Dr. Radu-Florin Damian, 2 Hours/Week, Specialization Section, Timetable:
Laboratory: Instructor: Assist.P. Dr. Radu-Florin Damian, 1 Hours/Week, Half Group, Timetable:

Evaluation
Type: **Examen**
A: 33.33%, (Test/Colloquium)
B: 33.33%, (Seminary/Laboratory/Project Activity)
D: 33.33%, (Homework/Specialty papers)

Grades
Not yet

Attendance
Not yet

Materials
Course Slides

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

Web Design

Concepte generale

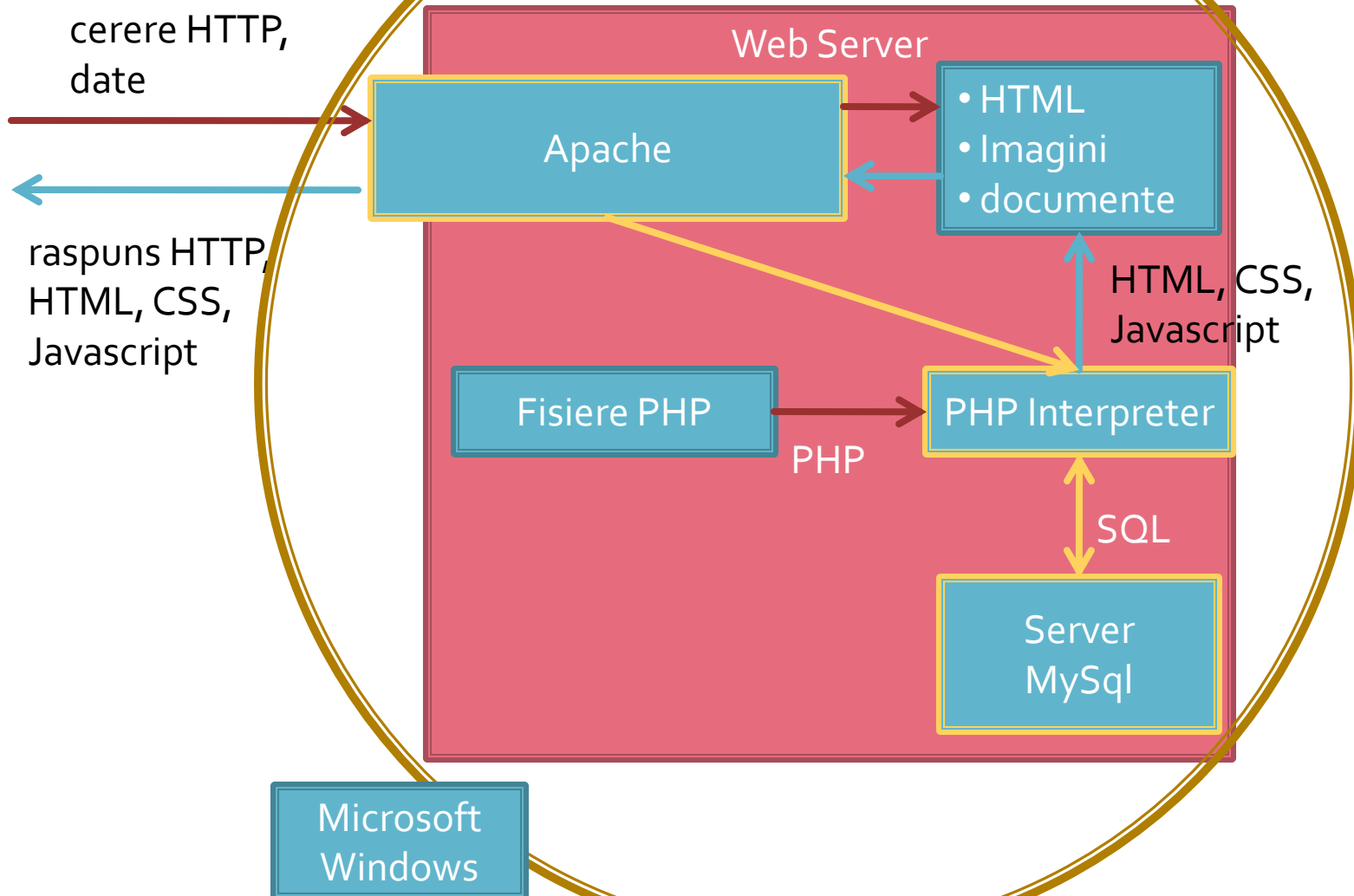
Concepte

- Steve Krug: “**Don't Make Me Think**”
- Utilizatorii scaneaza pagina, nu o citesc
- Informatia trebuie redusa la minimul necesar in majoritatea locurilor
- “Daca ceva e greu de utilizat, mai bine nu o utilizez”
- Utilizatorii au comportament de **rechin**
- Originalitatea **nu e** intotdeauna **recomandata**

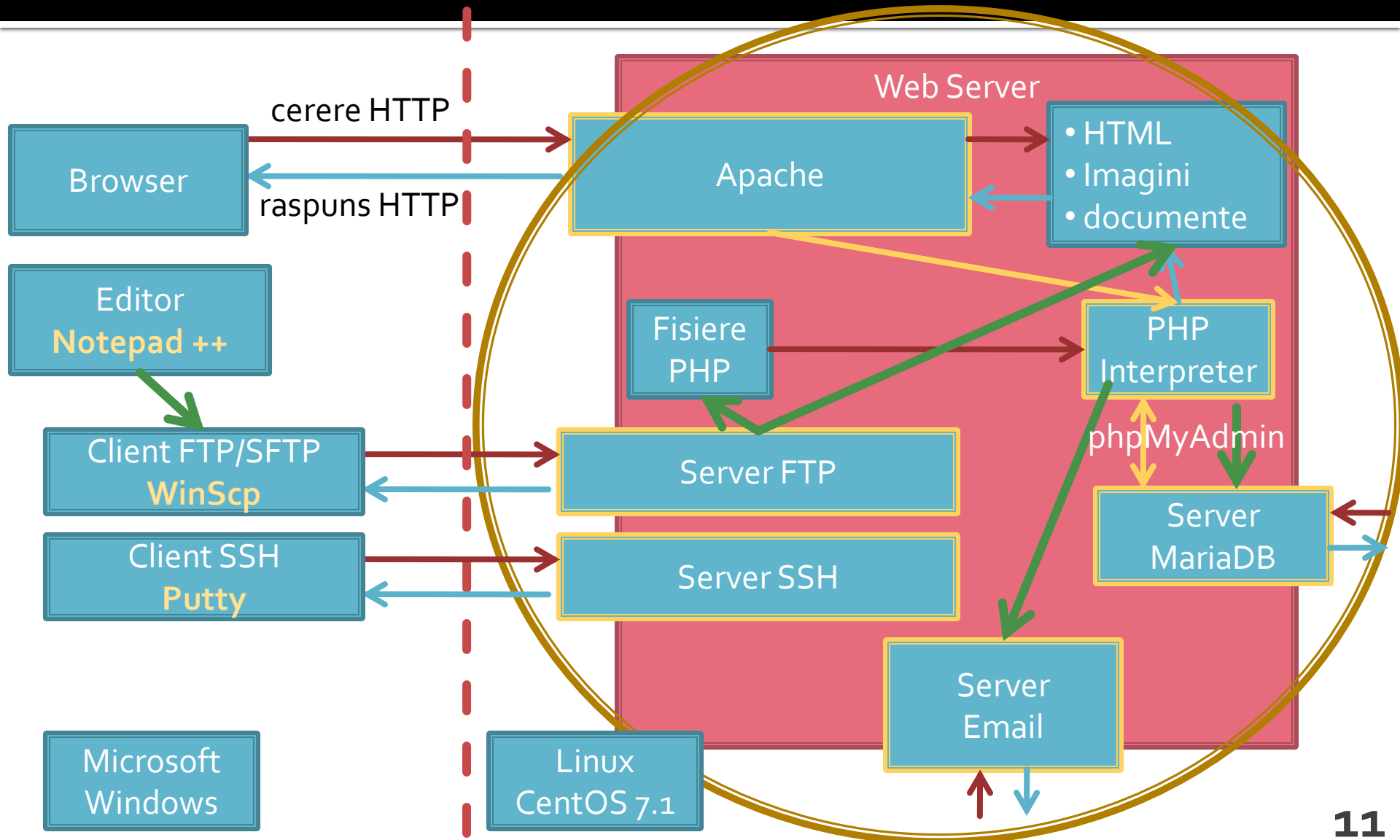
Consideratii generale

Aplicatii

WAMP



LAMP



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

- 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="Trimate" />
```

```
<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

Trimate



check1



but1



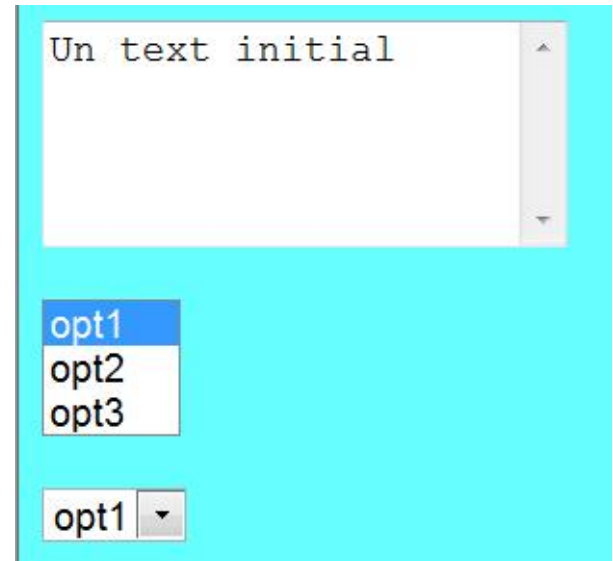
but2

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>
```



The image displays three visual representations of HTML form elements on a light blue background. At the top is a text area containing the text "Un text initial". Below it is a list box with three options: "opt1", "opt2", and "opt3", where "opt1" is selected and highlighted in blue. At the bottom is a dropdown menu with "opt1" selected and a small downward arrow on the right side.

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

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

```
539         <td><?php echo $row['Documente'];?>&nbsp;</td>
540         <td><?php echo $row['user_creat'];?>&nbsp;</td>
541         <td class="smaller"><a href="control_lot.php?id=<?php echo $row['ID_LOT'];?>">dezactiveaza</a><br /><a href=
"control_lot.php?id=<?php echo $row['ID_LOT'];?>">modifica</a></td>
542     </tr>
543     <?php $index++;
544     } while ( $row = mysql_fetch_assoc($result));?>
545 </table><?php
546 }
547 else
548 {
549     echo "<p>Nu exista loturi active</p>";
550 }
551 ?>
552
553 <p class="title">Loturi inactive</p>
554 <?php
555 $query = "SELECT l.*, c.`nume_user` AS `user_creat`
556         FROM `lot` AS l
557         LEFT JOIN `users` AS c ON (l.`User`=c.`id_user`)
558         WHERE l.`Activ` = 0 ORDER BY l.`ID_LOT` DESC";
559 $result = mysql_query($query);
560 $total=0;
561 if ($result && (mysql_num_rows($result) > 0))
562     {
563     $total=mysql_num_rows($result);
564     $row = mysql_fetch_assoc($result);
565     }
566 if ($total>0)
567 {?>
568 <table align="center">
569     <tr class="lista_titlu">
570         <td align="center">Nr. </td>
```

Separare cod PHP

- un fisier sursa PHP este un fisier HTML (in general) cu sectiuni de cod PHP
- interpretorul PHP cauta sectiunile pe care trebuie sa le interpreteze si interiorul lor proceseaza instructiuni ca fiind PHP
- ce se gaseste in **exteriorul** acestor sectiuni este trimis spre server-ul web **nemodificat**

Separare cod PHP

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

Variante de integrare

- echo afiseaza un text la "iesire" (echivalent cu printf() 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

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>';  
?>
```

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)

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

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"
 - `$_REQUEST` – ambele metode
- variabilele sunt **matrici** cu **cheia** data de atributul **name** din forma HTML
 - `<input type="text" name="carti_cant" size="3" maxlength="3" />`
 - `$_POST['carti_cant']` contine valoarea introdusa

Laborator 3

Laborator L3

- Sa se creeze un magazin simplu virtual care:
 - sa prezinte utilizatorului o lista de produse si preturi (constanta – maxim 5 produse)
 - sa preia de la acesta numarul de produse dorit
 - sa calculeze suma totala
 - sa adauge TVA 19%
 - sa prezinte un raport care sa contina:
 - total de plata
 - ora comenzii

Laborator L3 - continuare

- se creaza macar 3 pagini:
 - lista produse
 - formular comanda
 - rezultat
- forma paginilor:
 - tabel/CSS
- metoda
 - post
 - get

culoare	IMAGINE	culoare
	Continut (cu alta culoare fundal)	

Laborator – L3 - rezultat

Magazin online Firma X SRL

Lista Produse

Nr.	Produs	Pret
1	Carti	100
2	Caiete	50
3	Penare	150
4	Stilouri	125
5	Creioane	25

[Comanda](#)

Magazin online Firma X SRL

Realizati comanda

Nr.	Produs	Pret	Cantitate
1	Carti	100	<input type="text" value="1"/>
2	Caiete	50	<input type="text" value="2"/>
3	Penare	150	<input type="text" value="1"/>
4	Stilouri	125	<input type="text" value="0"/>
5	Creioane	25	<input type="text" value="0"/>

Magazin online Firma X SRL

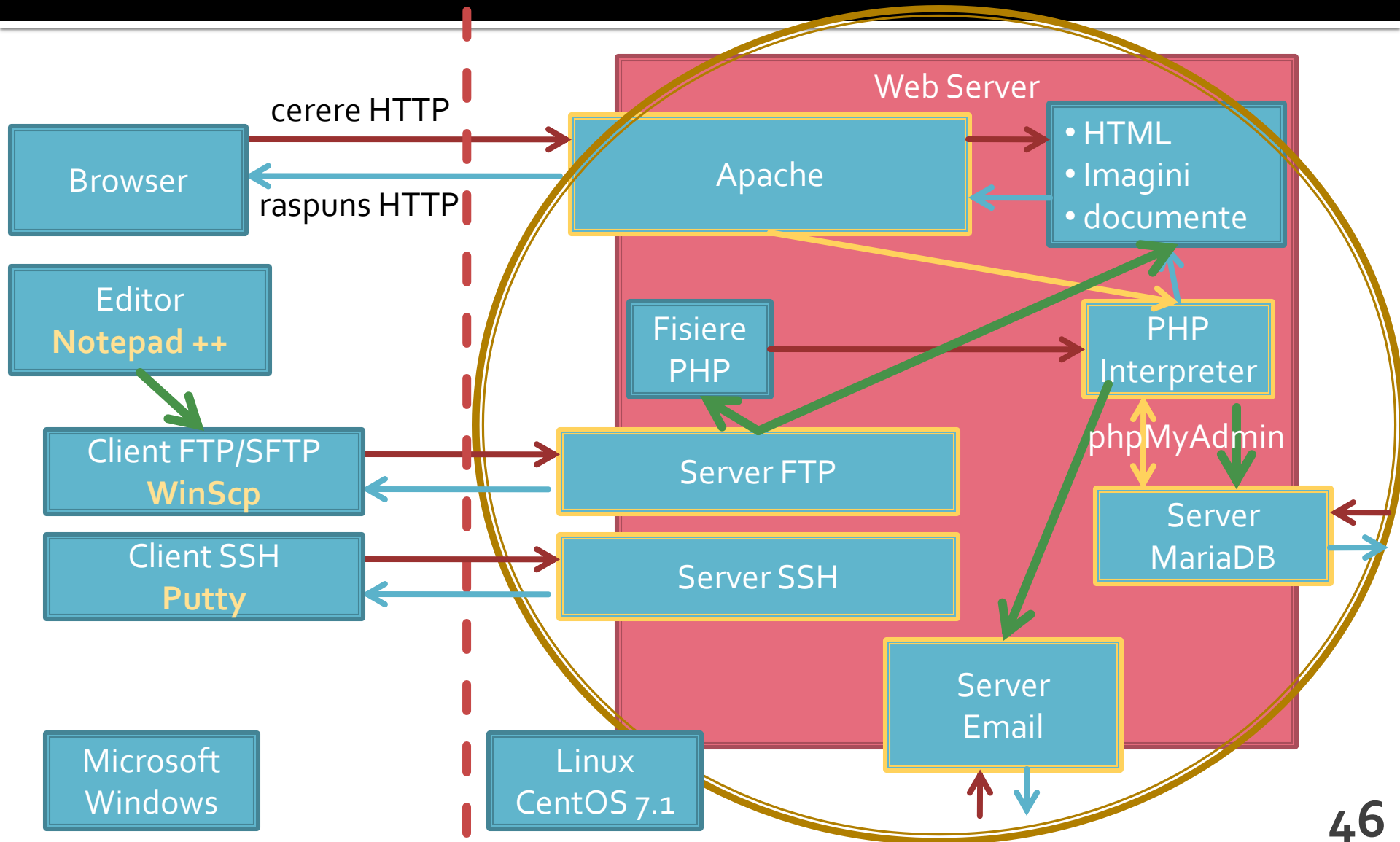
Rezultate comanda

Pret total (fara TVA): 350

Pret total (cu TVA): 416.5

Comanda receptionata la data: 17/03/2010 ora 08:24

Utilizare LAMP



Utilizare LAMP

Magazin online Firma X SRL

Lista Produse

Nr.	Produs	Pret
1	Carti	100
2	Caiete	50
3	Penare	150
4	Stilouri	125
5	Creioane	25

Comanda

Magazin online Firma X SRL

Realizati comanda

Nr.	Produs	Pret	Cantitate
1	Carti	100	<input type="text" value="1"/>
2	Caiete	50	<input type="text" value="2"/>
3	Penare	150	<input type="text" value="1"/>
4	Stilouri	125	<input type="text" value="0"/>
5	Creioane	25	<input type="text" value="0"/>

Trimite

Magazin online Firma X SRL

Rezultate comanda

Pret total (fara TVA): 350

Pret total (cu TVA): 416.5

Comanda receptionata la data: 17/03/2010 ora 08:24

`<input name="x" ..`

`$_POST['x']`

`$_GET['x']`

Web Server

Apache

Server FTP

Server SSH

Client / Server

```
<input name="nume" ....>
```

```
echo $_POST['nume']; //ceva  
echo $_GET['nume']; //ceva  
echo $_REQUEST['nume']; //ceva
```

ceva

Trimite

get
post

Interpretor PHP primeste
\$_POST
\$_GET
\$_REQUEST

Depanare

```
echo "<pre>";  
print_r($_POST);  
echo "</pre>";
```

```
<p>temp <?php echo  
"a=";echo $a; ?> </p>
```

Contact

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