

Laborator 3

2020/2021

Programarea aplicațiilor web

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

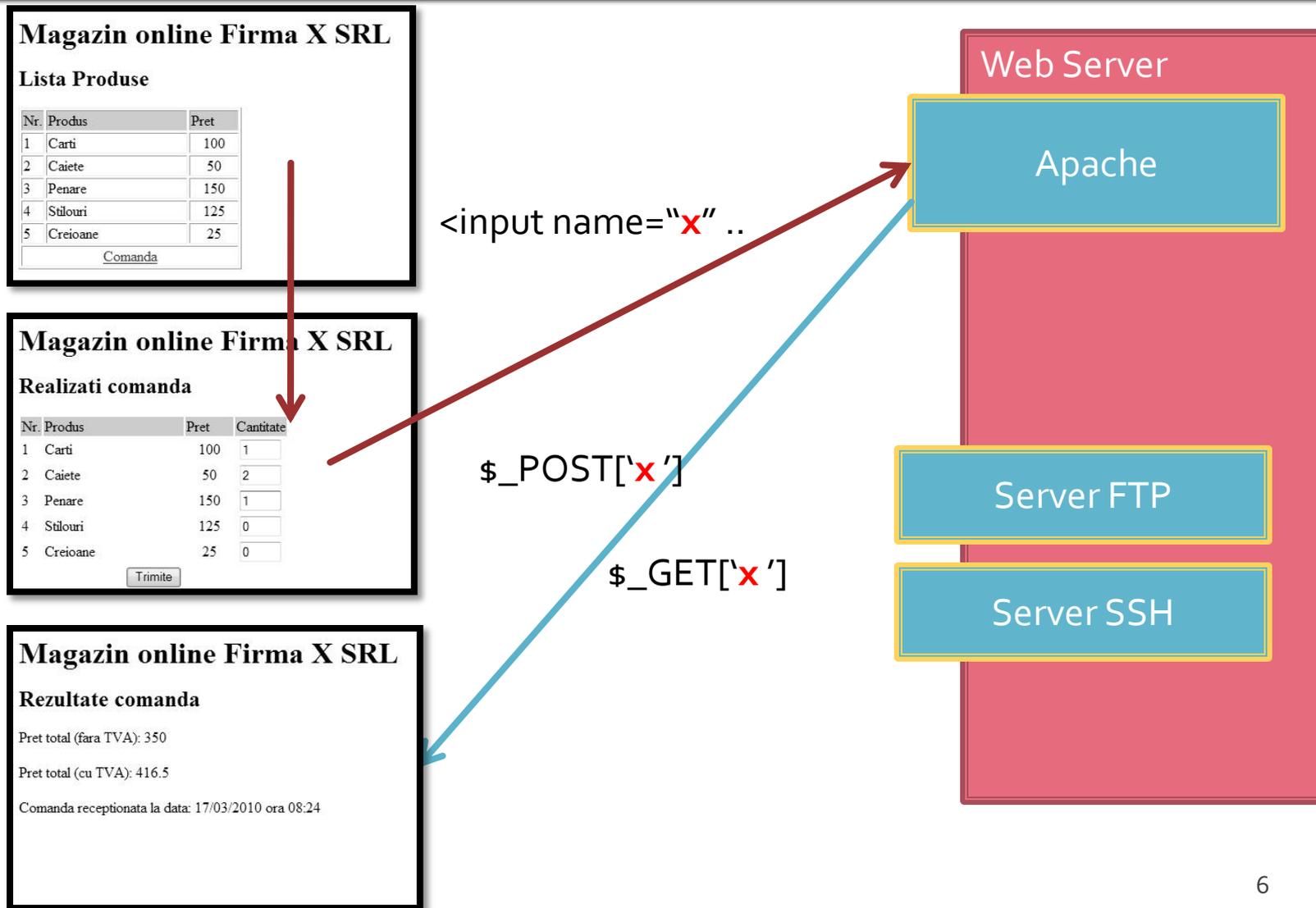
Rezultate comanda

Pret total (fara TVA): 350

Pret total (cu TVA): 416.5

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

Utilizare LAMP



Curs 2

Curs 2

- Online video
- “Examen” pentru prezenta curs
- PHP

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: HTML (+ CSS)

```
E:\Documents\Pagini Web\Safir dep\lista_lot.php - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Win
logfile.php lista_lot.php
539 <?php ?>
540
541 <div id="maincontent"><!-- InstanceBeginEditable name="Conti
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>Numar</td>
563 <td>Data</td>
564 <td>Nr. pui</td>
565 <td>Gr. pui</td>
566 <td>Nr. pui morti</td>
567 <td>Gr. pui morti</td>
568 <td>Ferma</td>
569 <td>Documente</td>
570 <td>Creat</td>
571 <td>Comenzi</td>
572 </tr>
573 <?php $index=1;
574 do { ?>
575 <tr class="<?php if ($index%2) echo "lista_i
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>
```

```
<table align="center">
<tr class="lista_titlu">
<td align="center">Nr.</td>
<td>Numar</td>
<td>Data</td>
<td>Nr. pui</td>
<td>Gr. pui</td>
```

Integrare: HTML + PHP

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>Numar</td>
563 <td>Data</td>
564 <td>Nr. pui</td>
565 <td>Gr. pui</td>
566 <td>Nr. pui morti</td>
567 <td>Gr. pui morti</td>
568 <td>Ferma</td>
569 <td>Documente</td>
570 <td>Creat</td>
571 <td>Comenzi</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>
```

```
<tr class="<?php if ($index%2) echo "lista_impar"; else echo "lista_p";
<td align="center"><?php echo $index; ?>&nbsp;</td>
<td><?php echo $row['Numar']; ?>&nbsp;</td>
<td><?php echo date("d/m/Y", strtotime($row['Data'])); ?>&nbsp;</td>
<td><?php echo $row['Pui']; ?>&nbsp;</td>
<td><?php echo $row['Greutate']; ?>&nbsp;</td>
```

```
<tr class="<?php if ($index%2) echo "lista_impar"; else echo "lista_p";
<td align="center"><?php echo $index; ?>&nbsp;</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

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

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
 - strict CSS

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

Exemplu



Analiza critica

- design?
 - in aplicatiile web forma **este importanta**
 - nu trebuie sa fie inovativa ci familiara
 - "Don't make me think!"

Design 1

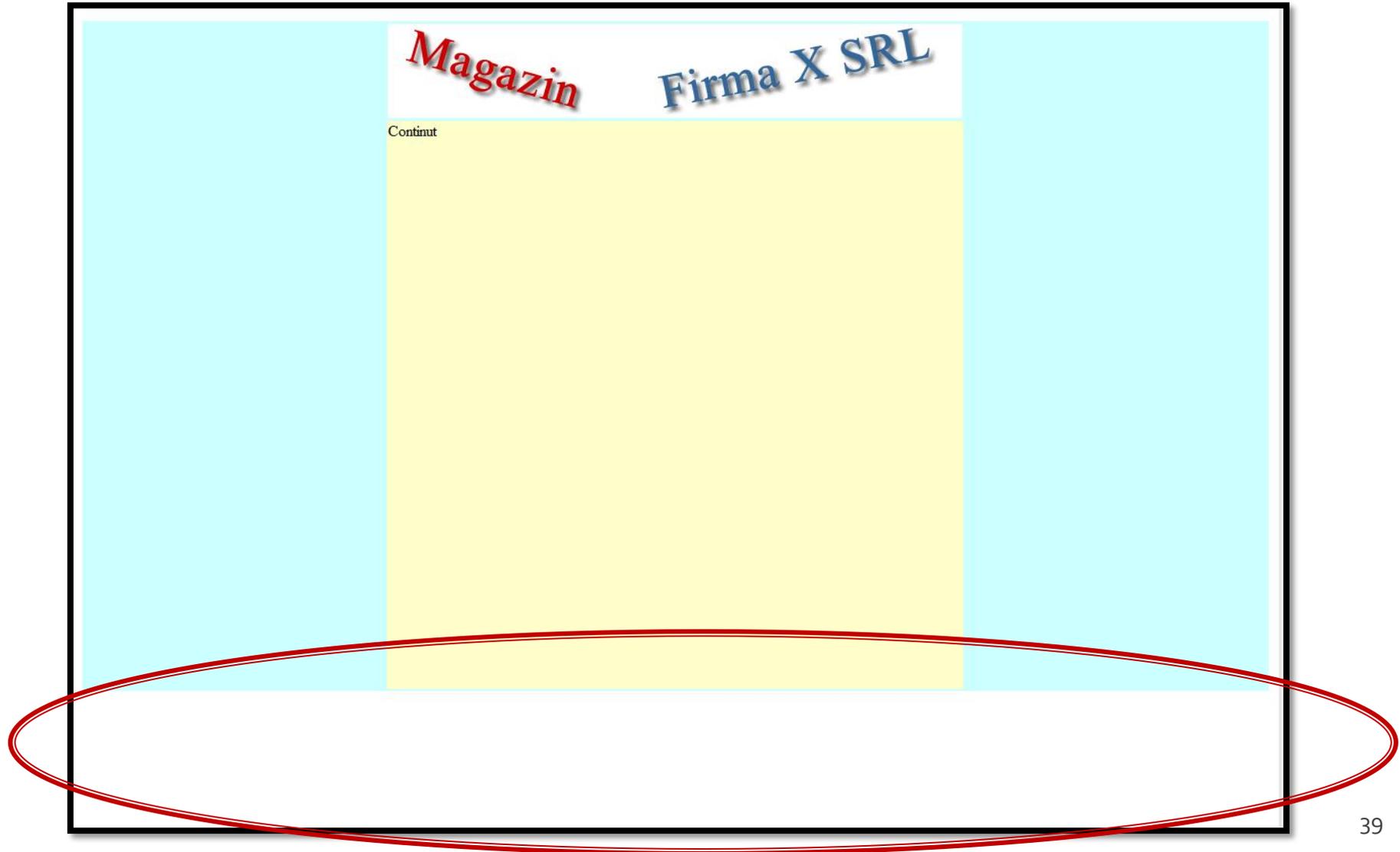
```
<html>
<head>
<title>Magazin online Firma X SRL </title>
</head>
<body>
<table width="100%" border="1"
bgcolor="#CCFFFF">
<tr><td rowspan="2" width="*"></td>
<td width="600"></td>
<td rowspan="2" width="*"></td></tr>
<tr><td height="600" valign="top"
bgcolor="#FFFACC">
Continut
</td></tr>
</table>
</body>
</html>
```

culoare	IMAGINE	culoare
	Continut (cu alta culoare fundal)	



tabel

Design 1



Design 2

```
<html>
<head>
<title>Magazin online Firma X SRL</title>
</head>
<body bgcolor="#CCFFFF">
<table width="600" border="0" align="center">
<tr><td></td></tr>
<tr><td height="600" valign="top"
bgcolor="#FFFCC">
Continut
</td></tr>
</table>
</body>
</html>
```

culoare	IMAGINE	culoare
	Continut (cu alta culoare fundal)	



tabel

Design 2

Magazin

Firma X SRL

Continut



Design 3

index.html

```
<html>
<head>
<title>Magazin online Firma X SRL</title>
<link rel="stylesheet" href="stil.css"
type="text/css" />
</head>
<body>
<div class="antet"></div>
<div class="continut">
Continut
</div>
</body>
</html>
```

stil.css

```
body { background-color: #CCFFFF;}
.antet { background-image:url(images/antet.gif);
background-repeat:no-repeat;
height: 100px;
width: 600px;
margin-top: 0px;
margin-right: auto;
margin-left: auto;}
.continut{ background-color:#FFFFCC;
height: 600px;
width: 600px;
margin-top: 5px;
margin-right: auto;
margin-left: auto;}
```

Design 3



Continuare

- capacitatea de extindere?
 - va apare un meniu?
 - unde?
- design 1
 - tabel cu 3 coloane, numai cea centrala e folosita
 - avantaj: se creaza doua zone care ar putea primi date **daca** e nevoie
 - dezavantaj: forma (culori, dimensiuni) intercalata in continut

Continuare

- design 3
 - avantaj: forma (culori, dimensiuni) separata de continut
 - avantaj: adaugarea altor cutii (div) care sa primeasca date **daca** e nevoie si controlul formei lor se poate realiza
 - dezavantaj: alaturarea a 2 box/div mai complexa, generatoare de probleme la incepatori
 - Bibliografie: "am mai facut ceva asemanator" – dezavantajul dispare

Laborator 3

Activitate laborator 3

- Prezenta si nota pentru activitatea de laborator se obtine **numai** prin completarea activitatii de examen online
 - perioada extinsa (1 saptamana)
 - necesar:
 - selfie (obligatoriu)
 - arhiva (zip/7zip) cu site-ul realizat: 3 fişiere sursă php + eventuale fisiere suport, de ex. imagini (opţional)

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

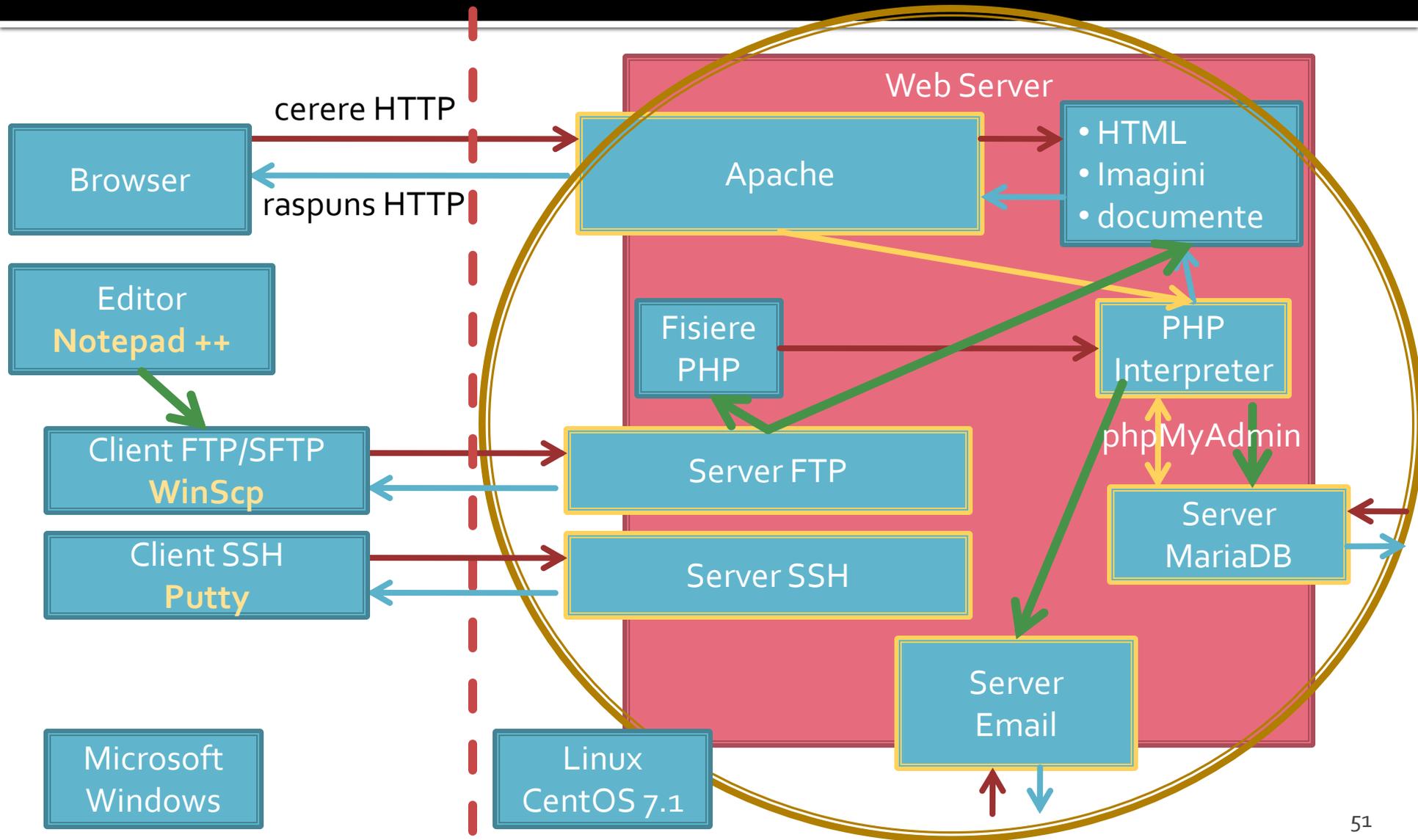
Rezultate comanda

Pret total (fara TVA): 350

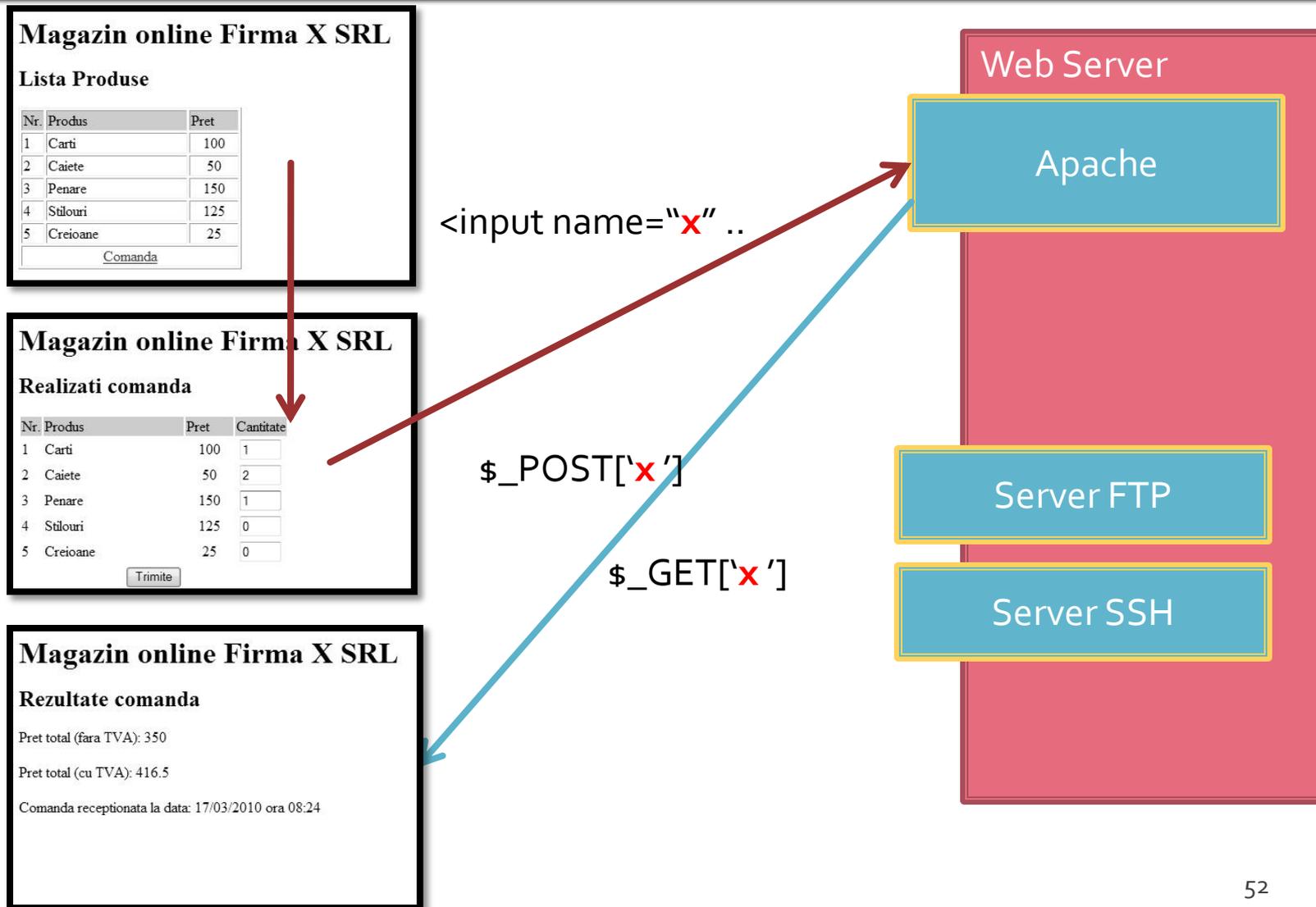
Pret total (cu TVA): 416.5

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

Utilizare LAMP

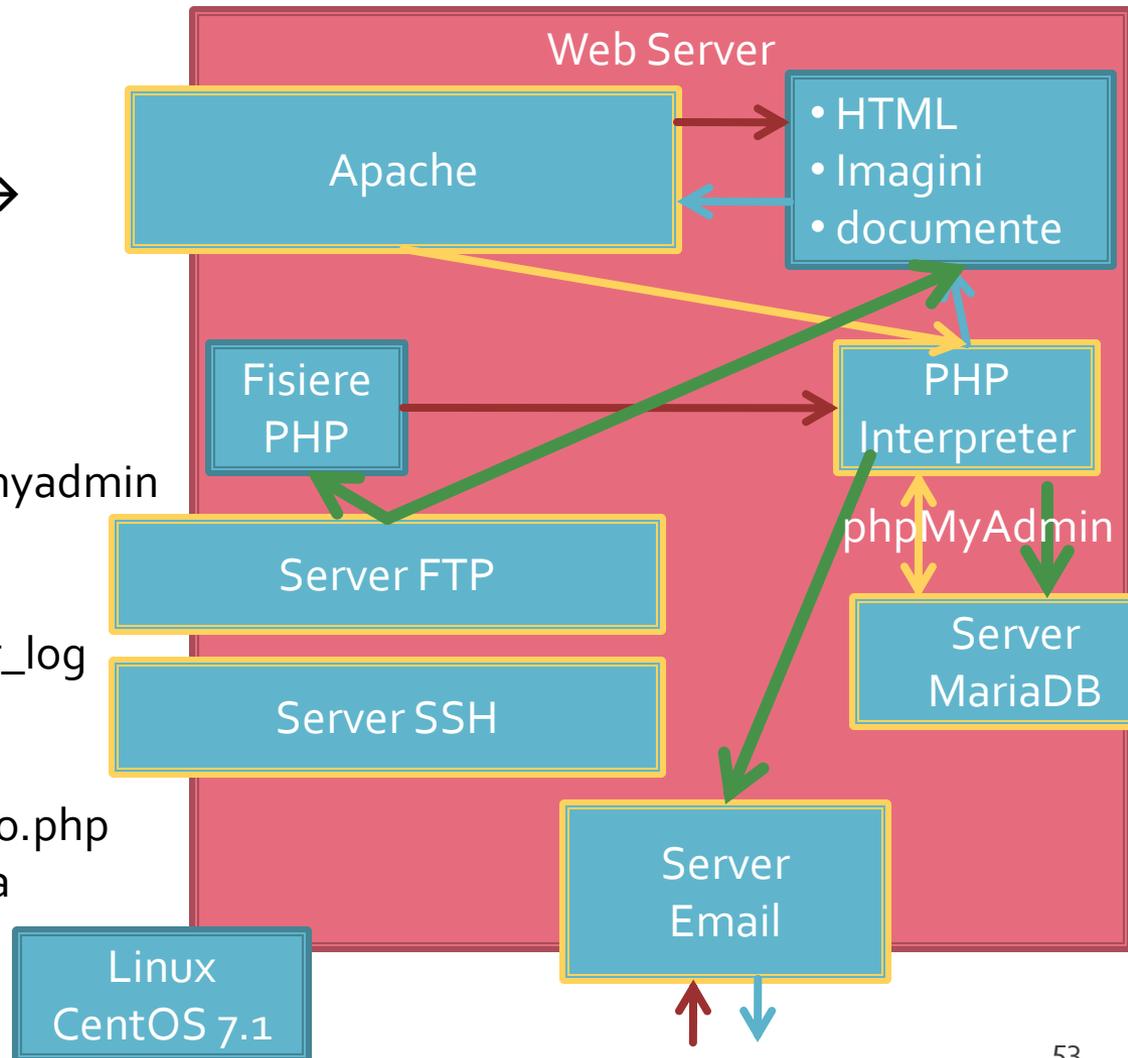


Utilizare LAMP



Utilizare LAMP

1. login → root:masterrc
2. ifconfig → 192.168.30.5
3. putty.exe → 192.168.30.5 → SSH → root:masterrc (remote login)
4. [alte comenzi linux dorite]
5. FTP → Winscp → SFTP → student:masterrc@192.168.30.5
6. MySql → http://192.168.30.5/phpmyadmin → root:masterrc
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
9. daca serviciul DHCP duce la oprirea Apache: `service httpd restart`



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