

Curs 11

2022/2023

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(17)-18 Curs
 - Luni 18-20 Laborator
 - Saptamanile 9-14
 - Luni 16(17)-18 Curs - **online**
 - Luni 18-20 Proiect - **online**

Orar

- <https://orar.etti.tuiasi.ro/> : C->16(17)-18, L/P -> 18



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										Damian R. PAW (C) 2.13 TC (R)	Damian R. PAW (L) 2.13 TC (R)	
Ma								C1	Scripcariu L. RCALSC (C)		Scripcariu L. RCALSC (L) 2.13 TC (R)	
Mi								Casian-Botez I. Etic (C) P6 (Amf.)	Casian-Botez I. Etic (S) P6 (Amf.)		Trifina L. TEFO (L) 3.25 TTI (L) Alecsandrescu I. POO (L) CI5(Corp C)	
J									Sirbu A. POO (C) P2 (Amf.)		Trifina L. TEFO (L) 3.25 TTI (L)	
V											Trifina L. TEFO (C) 3.25 TTI (L)	

Nota

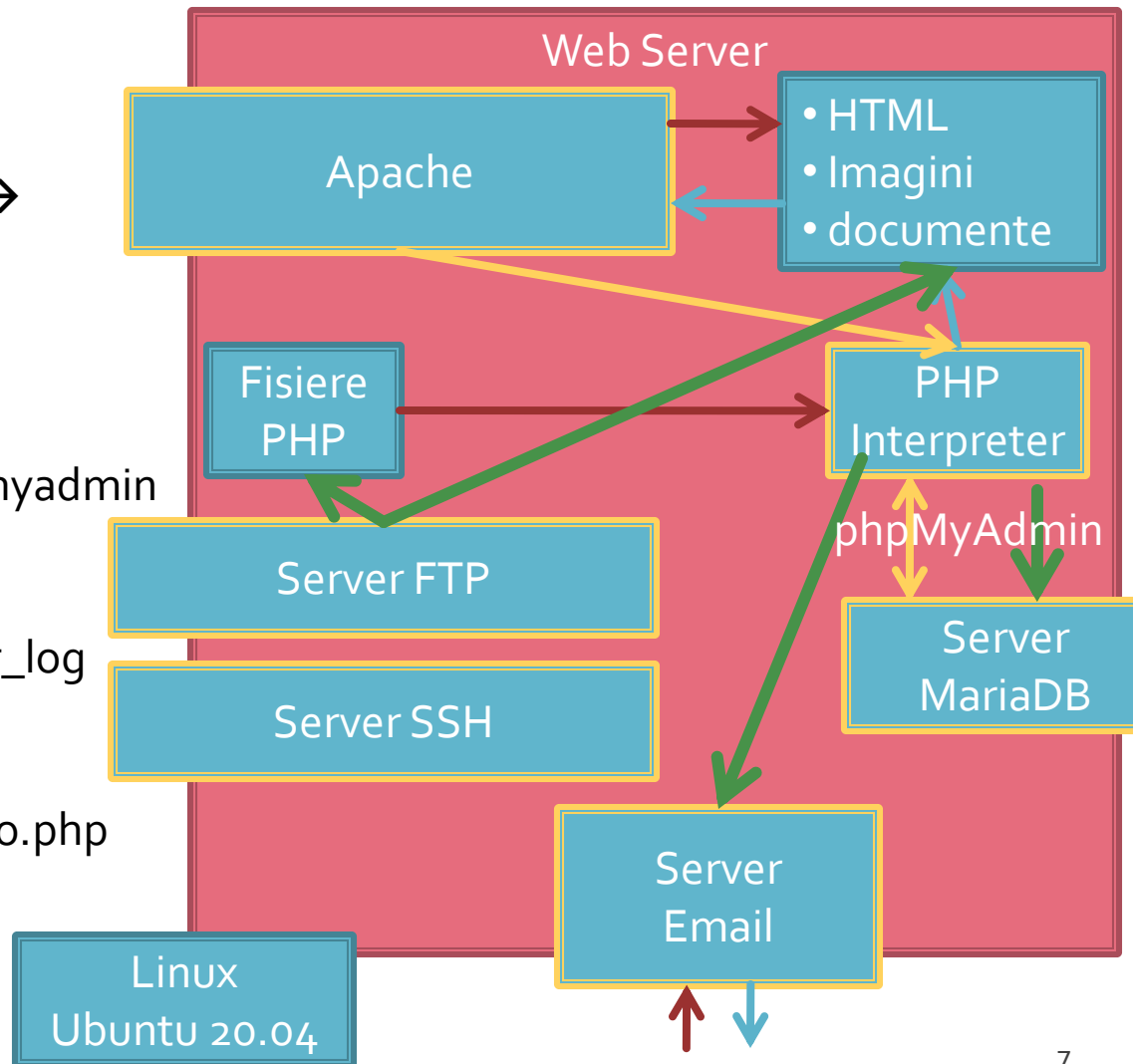
- An V
 - 33% E
 - 66% Aplicatii
 - 33% L
 - 33% P

Nota

- An V
 - 33% E
 - 33% L
 - 33% P
- Laborator - Prezenta
 - 1pz = 1p ($p > 5 \leftrightarrow$ Curs)
- Examen
 - Prezenta la curs: 3pz = 0.5p
 - Asemanator cu materia de proiect
- Activitate suplimentara
 - Dupa terminarea activitatii la laborator
 - +2 **1**p la E/L

Utilizzare LAMP

1. login → **paw**:masteretti
2. ifconfig → 192.168.30.5
3. putty.exe → 192.168.30.5 → SSH → **paw**:masteretti (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**:masteretti
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



Tema bonus

- logfile.php
 - Afiseaza log Apache (erori php majore)
- ~~1p 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 ✓
 - drepturi de acces la fisiere (Linux)

Server referinta LAMP

- Linux, doua variante
 - Centos 7.1
 - PHP 5.4.16
 - MariaDB 5.5.44 / root:masterrc
 - Apache 2.4.6
 - **root**/student:masterrc
 - **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 paw**:
 - **sudo usermod -a -G upload paw**
 - **sudo chmod -R 775 /var/www**

Project

Proiect

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

PROIECT (final)

- Tema de nota 8
 - Tema unica pentru fiecare student
 - Baza de date cu care se lucreaza contine minim 20 de inregistrari in tabelul cel mai "voluminos«
- Tema de nota 9
 - Conditiiile de la tema de nota 8 **si in plus**
 - Necesitatea conlucrarii intre 2 studenti cu doua teme "pereche"
 - Se accepta ca un student sa realizeze ambele puncte
 - Numar **minim** de pagini dinamice (php+mysql) in aplicatie **4 = 2 X 2**
 - Baza de date cu care se lucreaza contine minim 50 de inregistrari in tabelul cel mai "voluminos"

PROIECT (final)

- Tema de nota **10**
 - Condițiile de la tema de nota 9 **si in plus**
 - Necesitatea conlucrării între 2 studenti cu teme "pereche"
 - Tema se predă/trimite cu macar 1 zi înainte **sustinerii** ei
 - Numar **minim** de pagini dinamice (php+mysql) in aplicatie **6 = 3 X 2**
 - Baza de date cu care se lucreaza sa contina minim 100 de inregistrari in tabelul cel mai "voluminos".

PROIECT (final)

- Tema de nota **10+**
 - Condițiile de la tema de nota 10 **si in plus**
 - Numar **minim** de pagini dinamice (php+mysql) in aplicatie **8 = 4 X 2**
 - Baza de date cu care se lucreaza contine minim **300** de inregistrari in tabelul cel mai "voluminos"
 - Necesitatea investigarii posibilitatilor de **imbunatatire** a aplicatiei si adaugarii de functionalitate (**obligatoriu**)
 - nota individuala la proiect va depinde intr-o mica masura (in limita a 1p) de nota minima a colegilor din echipa
 - **+1p la nota de examen**

PROIECT (final)

- In caz de necesitate, pentru completarea echipei cadrul didactic poate fi membru al echipelor (9/10/10+). Conditii:
 - metoda de comunicare in echipa sa fie prin email sau direct
 - latenta de raspuns: ~ 1 zi
 - reactiv
 - nota implicita 10 (😊)
 - nu lucreaza noaptea, si in special nu in noaptea dinaintea predarii (😊)
- dezavantaj asumat: "spion" in echipa

PROIECT (final)

- Tema bonus (>5, in general **offline**)
 - Conditiiile de la tema de nota 10+ **si in plus:**
 - **3 studenti/CD**
 - Baza de date cu care se lucreaza contine minim **500** de inregistrari in tabelul cel mai "voluminos"
 - Numar **minim** de pagini dinamice (php+mysql) in aplicatie **15 = 5 X 3**
 - Tema care face apel la controlul **sesiunii** client/server
 - Necesitatea utilizarii **Javascript** in **aplicatie** (aplicatie libera dar cu efect tehnic nu estetic)
 - Forma paginii controlata dual prin CSS, desktop/phone
 - Facilitati in ceea ce priveste nota (**DACA** toate celelalte conditii sunt indeplinite), la alegere:
 - prezenta la laborator: N → P = **66%**, L = **0%**, E = 33%
 - **+2p la nota de examen**

PROIECT (final)

- proiectul se **sustine individual** (oral si practic)
- fiecare membru al unei echipe (la teme de nota 9 si 10) trebuie sa sustina in aceeasi zi proiectul
- nota individuala la proiect va depinde intr-o mica masura (in limita a 1p) de nota medie a colegilor din echipa (numai la teme de 10+)
 - $N-\min(E)=1 \rightarrow -0.5 p$
 - $N-\min(E)=2 \rightarrow -1 p$
 - $N-\min(E)=3 \rightarrow -1.5 p$

Notare proiect 2020/2021

- 1p – functionalitate ✓
- 1p – aplicatia ruleaza pe server-ul CentOS/Ubuntu ✓
- numar de pagini dinamice ✓
- numar de inregistrari in baza de date ✓
- 1p – planul aplicatiei ✓
- 2p – prezentare in Teams a proiectului ✓

Notare 2016

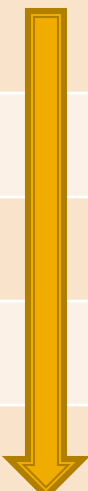
- 1p – functionalitate
- 1p – mutarea **personală** a site-ului (restaurare backup + setare server) pe un server de referință
 - server-ul de referință va fi mașina virtuală **Centos 7.1** utilizată la laborator (inclusiv aplicațiile cu pricina)
 - să vă pregătiți pentru situația în care pe acel server există și alte baze de date care **nu** trebuie distruse
 - fiecare student își pune sursele în directorul propriu, în rădăcina server-ului. Dacă tema depinde de anumite fișiere ale colegului, le cereți înainte
- 1p – cunoașterea codului
 - răspunsul la întrebări de genul: “unde ai făcut aceasta”
- Teme “de nota 10,10+”
 - inițiativă. Investigarea posibilităților de îmbunătățire
 - întrebări legate de cooperarea cu colegul de echipă
 - explicații relativ la funcționarea unei anumite secvențe de cod
 - utilizare sesiune, Javascript, F shape pattern (CSS media)

Notare 2023 (final)

- 1p – **functionalitate**
- 1p – mutarea **personală** a site-ului (restaurare backup + setare server) pe un server de referință CentOS/**Ubuntu**
- 1p – cunoasterea **codului**
 - răspunsul la întrebări de genul: "unde ai făcut aceasta", "ce face acest cod"
- 1p – **planul aplicației**
- Teme "de nota 10,10+"
 - Inițiativa. Investigarea posibilităților de îmbunătățire
 - Explicații relativ la funcționarea unei anumite secvențe de cod
 - Utilizare sesiune, Javascript, **CSS media**

Notare 2022

- numar de pagini dinamice ✓
- numar de inregistrari in baza de date ✓
 - se verifica indeplinirea conditiilor corespunzatoare si se realizeaza **de-clasificarea** temei pana cand **ambele** conditii sunt indeplinite

Tema de nota ...		Pagini	Inregistrari
	bonus	$15 = 5 \times 3$	500
	10+	$8 = 4 \times 2$	300
	10	$6 = 3 \times 2$	100
	9	$4 = 2 \times 2$	50
	8	$1 = 1 \times 1$	20

Exemplu

- 1. Galerie de imagini in care imaginile sunt ordonate dupa categorii.

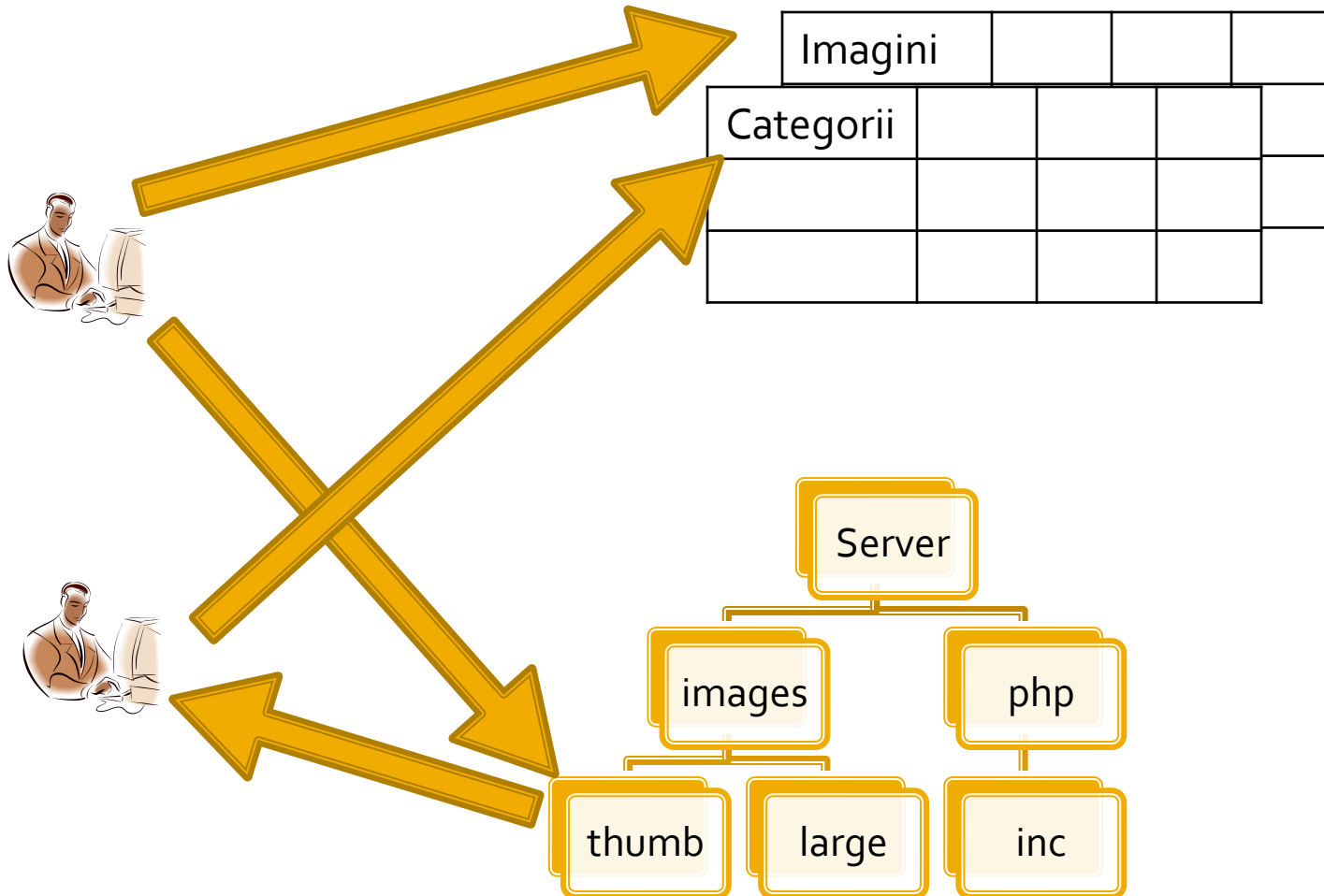


- a. aplicatia pentru adaugarea de categorii si afisare a imaginilor (cu alegerea prealabila a categoriei si afisarea listei de imagini format mic)



- b. aplicatia pentru adaugare de imaginilor (cu alegerea prealabila a categoriei si generarea prealabila a imaginii format mic)

Exemplu



“Examen” Alocare teme proiect

- **Alocare teme**

- tema aleasa (optiune principala) - **necesar**
- nume coechipier - **necesar**
- tema alternativa (rezerva 1)
- tema alternativa (rezerva 2)
- punctul ales (a/b) - **necesar**

- Primul venit, primul servit

- **ambii** parteneri finalizeaza examenul

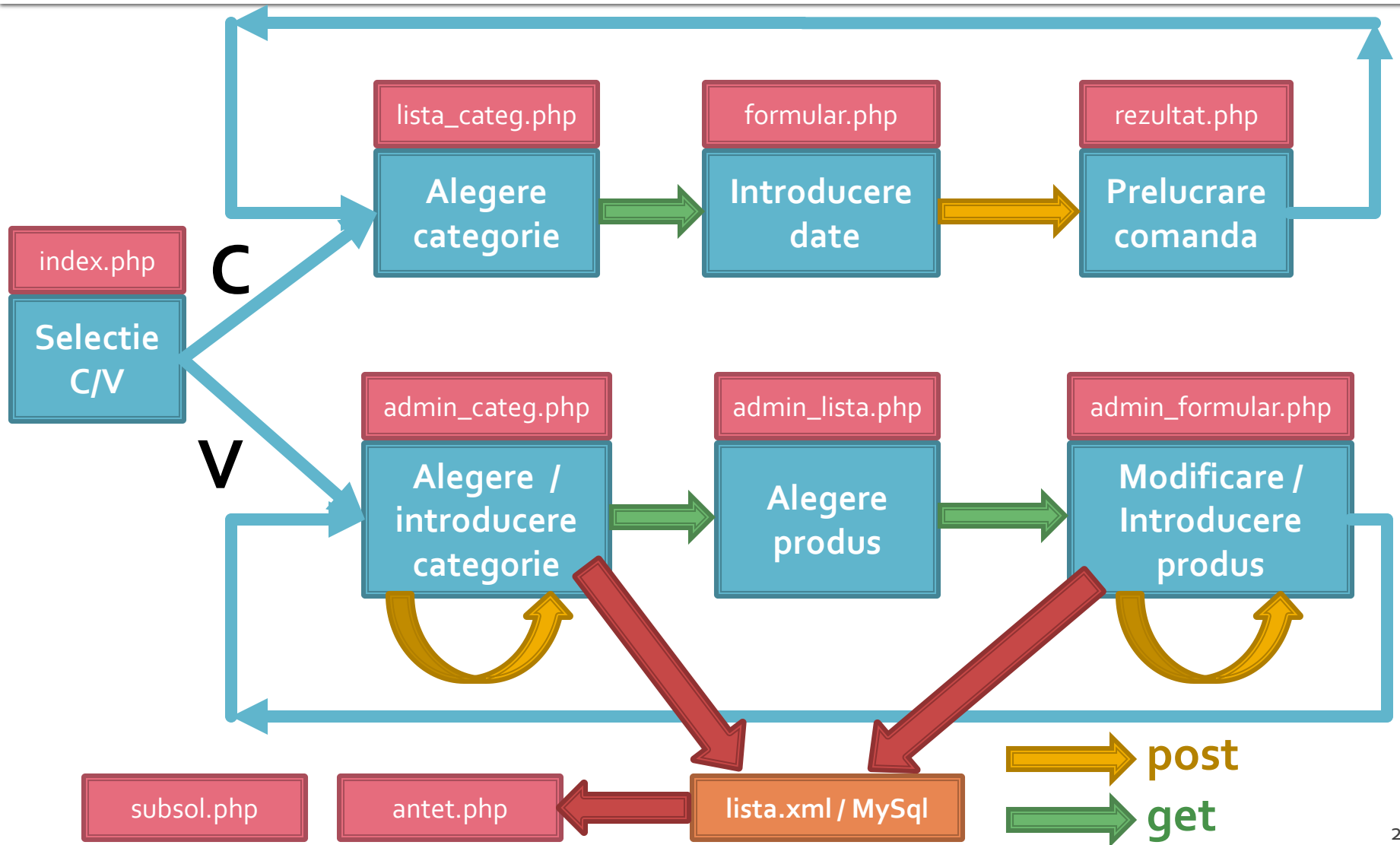
“Examen” Predare proiect

- Predare proiect
- Chiar daca unele fisiere sunt comune, **ambii** coechipieri trebuie sa finalizeze depunerea
- Predare 3 fisiere
 - un fișier ***.pdf/*.jpg** cu **planul aplicației**
 - un fișier ***.sql** cu backup-ul bazei de date de care are nevoie aplicația pentru a funcționa (**nr. linii!!**)
 - un fișier cu arhiva directorului conținând aplicația (fișiere *.php, *.jpg, structură de directoare etc., arhivate: ***.zip, *.7z** etc.) (**nr. pagini!!**)

Server referinta LAMP

- Linux, doua variante
 - Centos 7.1
 - PHP 5.4.16
 - MariaDB 5.5.44 / root:masterrc
 - Apache 2.4.6
 - **root**/student:masterrc
 - **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 paw**:
 - **sudo usermod -a -G upload paw**
 - **sudo chmod -R 775 /var/www**

Plan aplicatie



Examen

- **fizic**
- probleme
- fiecare student are subiect propriu
- toate materialele permise
- tehnica de calcul **nu** este necesara dar este permisa

Examen

- Oricare din temele de proiect (sau asemenea) poate constitui una din problemele de examen
 - se va cere realizarea planului / structurii logice a aplicatiei
- Se poate cere scrierea unui cod pentru realizarea anumitor operatii, fara necesitatea corectitudinii tehnice absolute (";", nume corect al functiilor, parametri functie etc.)
- Se poate cere interpretarea unui cod php/MySql cu identificarea efectului

Aspecte practice recomandate in realizarea aplicatiilor web

Metode de lucru recomandate 1

- Daca nu aveti acces simplu la “log-urile” server-ului **MySQL** puteti vedea cum ajung efectiv interogariile la el afisand temporar textul interogarii
 - `$query = "SELECT * FROM `produse` AS p
WHERE `id_categ` = ".$row_result_c['id_categ'];
echo $query; //util in perioada de testare`
 - Textul prelucrat de PHP al interogarii va fi afisat in clar pe pagina facand mai usoara depanarea programului
 - Aceste linii **trebuie** eliminate in forma finala a programului ca masura de securitate

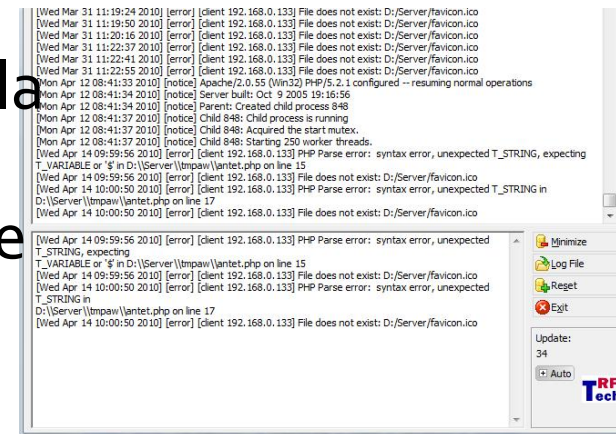
Metode de lucru recomandate 2

- Verificarea “log-ului” de erori al server-ului Apache ramane principala metoda de depanare a codului PHP.

- W2000: Utilizarea aplicatiei prezentata la laborator este mai comoda datorita automatizarii dar orice alta varianta este utila

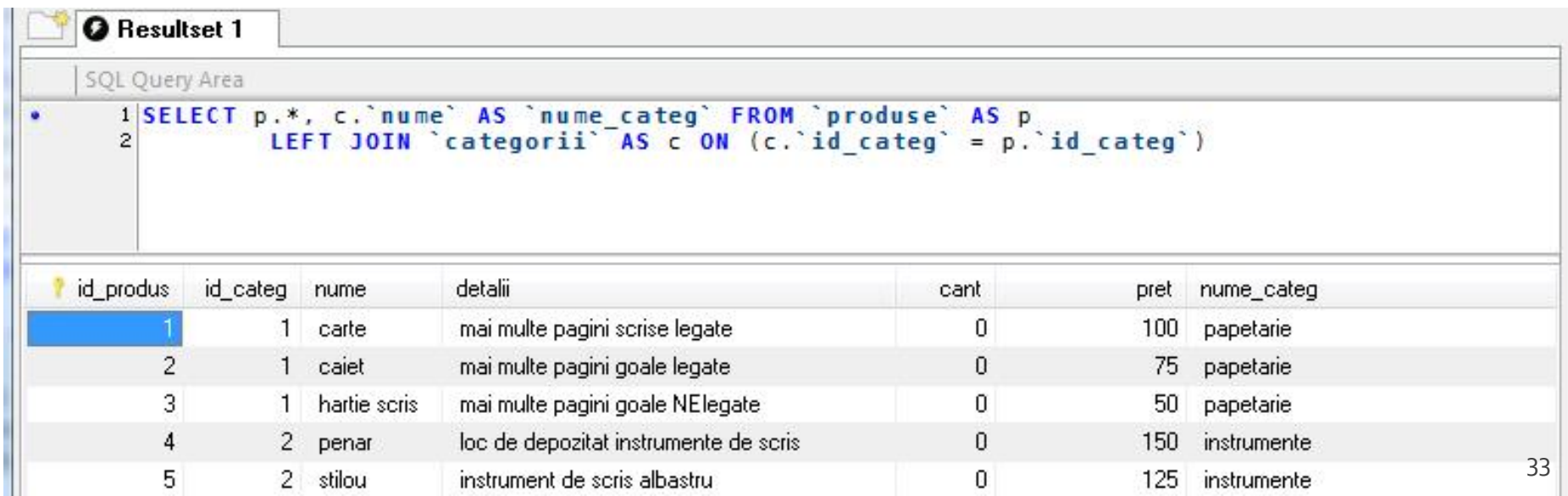
- Centos 7.1:

- putty → nano /var/log/httpd/error_log
- <http://192.168.30.5/logfile.php> (nonstandard)
- tema suplimentara (php.ini + log PHP **recomandat**)



Metode de lucru recomandate 3

- In perioada de definitivare a formei interogarilor MySql este de multe ori benefic sa se utilizeze mai intai **MySql Workbench/PhpMyAdmin** pentru incercarea interogarilor, urmand ca apoi, cand sunteti multumiti de rezultat, sa transferati interogarea SQL in codul PHP



The screenshot shows the MySQL Workbench interface. At the top, there's a tab labeled 'Resultset 1'. Below it is the 'SQL Query Area' containing the following SQL query:

```
1 SELECT p.*, c.`nume` AS `nume_categ` FROM `produse` AS p
2 LEFT JOIN `categorii` AS c ON (c.`id_categ` = p.`id_categ`)
```

Below the query area, the results are displayed in a table with the following columns: id_produș, id_categ, nume, detalii, cant, pret, and nume_categ. The table contains 5 rows of data.

id_produș	id_categ	nume	detalii	cant	pret	nume_categ
1	1	carte	mai multe pagini scrise legate	0	100	papetarie
2	1	caiet	mai multe pagini goale legate	0	75	papetarie
3	1	hartie scris	mai multe pagini goale NElegate	0	50	papetarie
4	2	penar	loc de depozitat instrumente de scris	0	150	instrumente
5	2	stilou	instrument de scris albastru	0	125	instrumente

Metode de lucru recomandate 3

MySQL Query Browser - Connection: root@server / tmpaw

File Edit View Query Script Tools Window Help

Transaction Explain Compare

Resultset 1

SQL Query Area

```
1 SELECT p.*, c.`nume` AS `nume_categ` FROM `produse` AS p
2 LEFT JOIN `categorii` AS c ON (c.`id_categ` = p.`id_categ`)
```

id_produs	id_categ	nume	detalii	cant	pret	nume_categ
1	1	carte	mai multe pagini scrise legate	0	100	papetarie
2	1	caiet	mai multe pagini goale legate	0	75	papetarie
3	1	hartie scris	mai multe pagini goale NElegate	0	50	papetarie
4	2	penar	loc de depozitat instrumente de scris	0	150	instrumente
5	2	stilou	instrument de scris albastru	0	125	instrumente
6	2	creion	instrument de scris gri	0	25	instrumente
7	3	cd	canta	0	50	audio-video
8	3	dvd	vizual	0	100	audio-video
9	3	blue ray	vizual extrem	0	500	audio-video

9 rows fetched in 0.0035s (0.0016s)

Edit Apply Changes Discard Changes First Last Search

1: 1

Metode de lucru recomandate 4

- eficienta unei aplicatii web
 - 100% - toate prelucrarile "mutate" in RDBMS
 - PHP doar afisarea datelor
- eficienta unei aplicatii MySql
 - 25% alegerea corecta a tipurilor de date
 - 25% crearea indecsilor necesari in aplicatii
 - 25% normalizarea corecta a bazei de date
 - 20% cresterea complexitatii interogarilor pentru a "muta" prelucrarile pe server-ul de baze de date
 - 5% scrierea corecta a interogarilor

Metode de lucru recomandate 5a

- La implementarea unei aplicatii noi (proiect)
 1. Imaginarea planului aplicatiei (ex: S25)
 - "cum as vrea eu sa lucrez cu o astfel de aplicatie"
 - hartie/creion/timp – esentiale
 2. Identificarea datelor/transmisia de date intre pagini
 - get/post/fisier unic colectare-prelucrare
 - baza de date read/write
 3. Identificarea structurii logice a datelor utilizate
 - "clase" de obiecte/fenomene tratate identic
 - se are in vedere scalabilitatea (posibilitatea de crestere a numarului de elemente dintr-o clasa)

Metode de lucru recomandate 5b

- La implementarea unei aplicatii noi (proiect)
 - 4. Realizarea structurii bazei de date
 - In general un tabel pentru fiecare clasa logica distincta **DAR...**
 - se are in vedere scalabilitatea (daca aplicatia creste sa **NU** apara cresterea numarului de clase/tabele) **SI...**
 - normalizare
 - 5. Identificarea tipului de date necesar pentru coloane
 - de preferat numerele intregi in orice situatie care presupune ordonare
 - dimensiunea campurilor nu mai mare decat e necesar (poate fi fortata prin atributul "size" in eticheta HTML "input")
 - 6. Imaginarea forme fizice a paginilor
 - "am mai vazut asa si mi-a placut" (Don't make me think!)
 - investigarea posibilitatii de a introduce functionalitate template

Metode de lucru recomandate 5c

- La implementarea unei aplicatii noi (proiect)
 - 7. Popularea manuala a bazei de date cu date initiale
 - MySql Query Browser (sau PhpMyAdmin) / automat / imprumut
 - programarea individuala a paginilor are nevoie de prezenta unor date
 - 8. Programare individuala a paginilor
 - In general in ordinea din planul aplicatiei (de multe ori o pagina asigura datele necesare pentru urmatoarea din plan)
 - modul "verbose" activ pentru PHP (adica: `echo $a; print_r($matr)`)
 - 9. Pregatirea pentru distributie/mutare
 - testare detaliata (eventual un "cobai")
 - eliminarea adaosurilor "verbose"
 - backup
 - generarea unui eventual install/setup

Tehnici PHP avansate

HTTP headers

- Permite transmiterea unor header-e specifice protocolului HTTP
- Structura mesajului
 - <initial line, different for request vs. response>
 - Header1: value1
 - Header2: value2
 - Header3: value3
 -
 - <optional message body goes here, like file contents or query data; it can be many lines long, or even binary data \$&*%@!^\$@>

HTTP headers

- header(string, code)

```
<?php header("HTTP/1.0 404 Not Found");?>
```

```
<?php header("Location: http://www.example.com/");  
/* Redirect browser */?>
```

```
<meta http-equiv="refresh" content="5;  
url=http://www.example.com/">
```

HTTP headers

- Header-ele HTTP se trimit inaintea oricaror alte date (HTML)
 - Inceput fisier: `<?php header("..."); ?><!DOCTYPE HTML PUBLIC ...
<html>...<body>...</body></html>`
 - Nici macar **un spatiu** nu trebuie sa apara inainte de primul `<?php`
 - Daca necesitatea de a trimite header-e poate aparea mai tarziu in script se foloseste obligatoriu `Buffer ieseire`

Buffer iesire

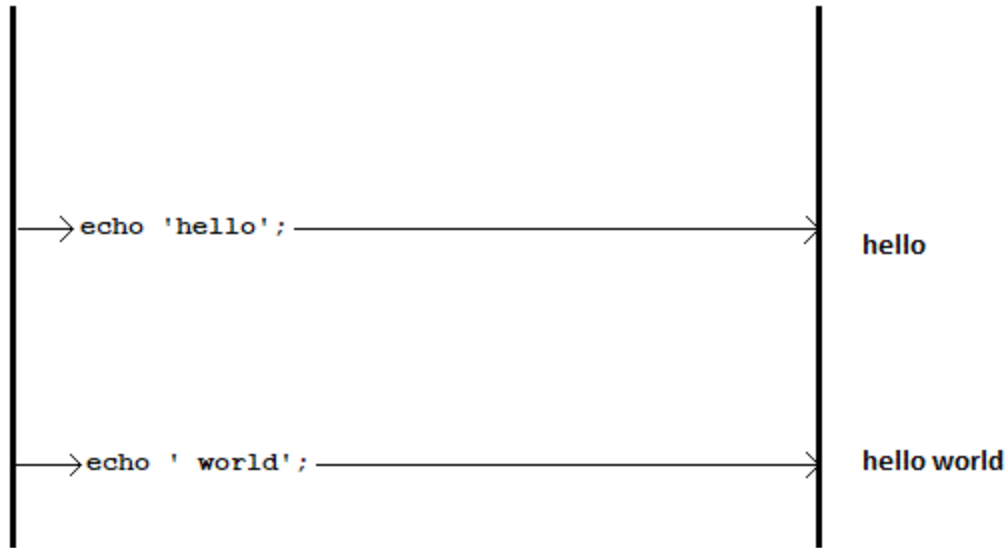
- Copie orice iesire a scriptului PHP intr-un buffer de memorie fara sa transmita nimic clientului
- Utilizat in general pentru conlucrarea cu header-e HTTP, evitarea generarii de HTML inainte de terminarea lucrului cu header-e
- `ob_start();`
- `ob_end_flush ();`
- `ob_end_clean ();`
- `ob_get_contents ()`

Buffer issue

No output buffering

PHP script

Client Browser

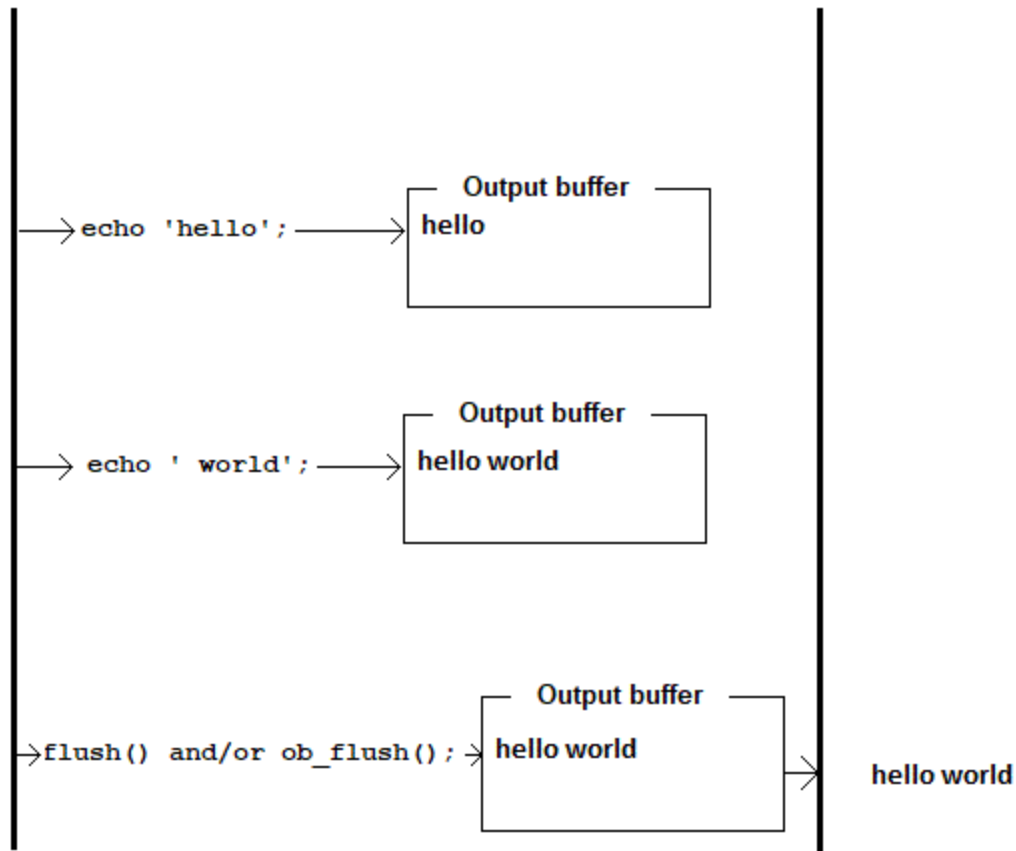


Buffer issues

Output buffering

PHP script

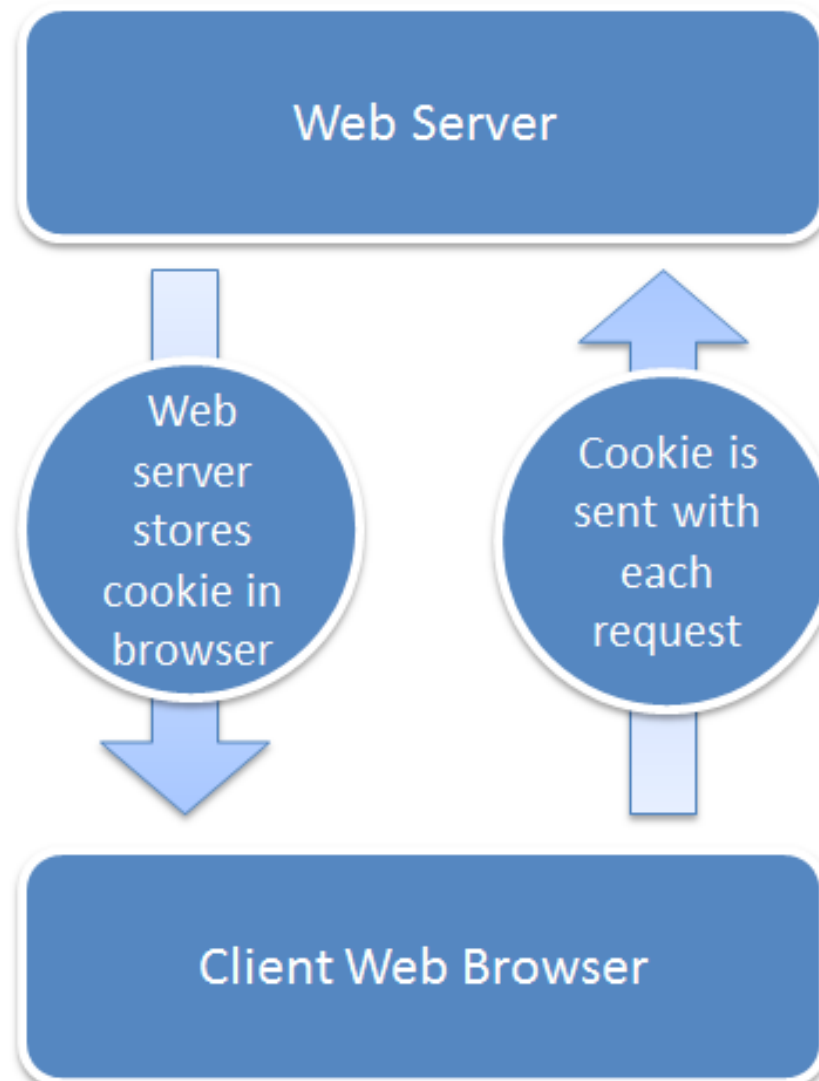
Client Browser



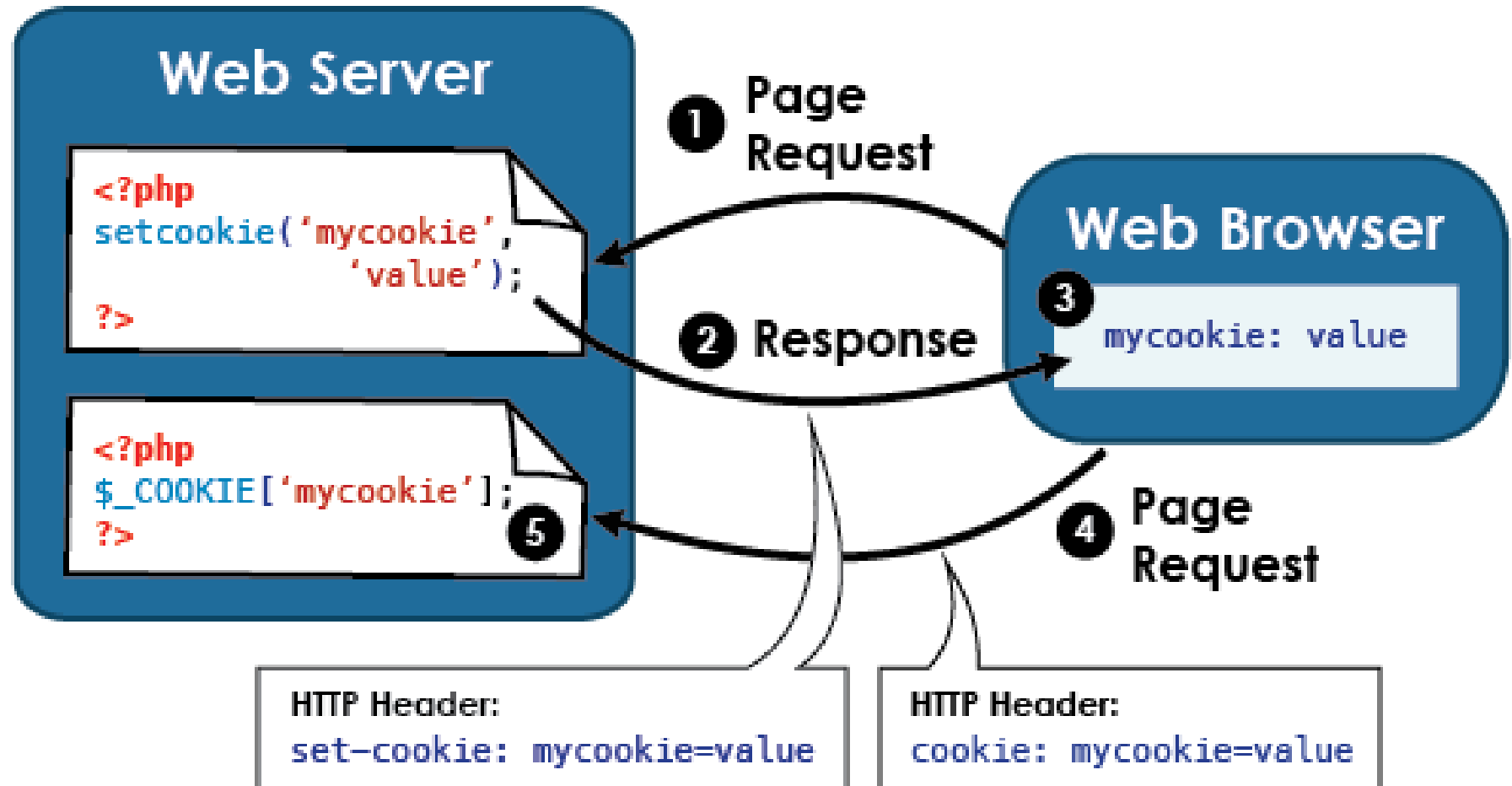
Cookies

- mici cantitati de date ce se stocheaza pe masina client (de obicei gestionat de browser)
- Circula impreuna cu (**este**) header HTTP
- setcookie (string name , string value , int expire , string path , string domain , bool secure , bool httponly)
 - nume (ptr. identificare)
 - value (valoarea/datele stocate)

Cookies



Cookies



Cookies

- `setcookie(string $name, string $value , int $expire = 0)`
 - `expire`: UNIX time stamp, nr. sec. din 1970
 - `time()+nr. sec. de viata dorite`
- datele se stocheaza pe client: probleme de securitate
- Se poate obtine valoarea memorata prin variabila globala `$_COOKIE['nume']`
 - **NU** in acelasi script
 - daca un script php trimite un cookie cu header-ele, de-abia **urmatorul** script accesat va primi acele cookie in header-e

Cookies

```
<?php
$value = 'something from somewhere';

setcookie("TestCookie", $value);
setcookie("TestCookie", $value, time()+3600); /* expire in 1
hour */
setcookie("TestCookie", $value, time()+3600, "/~rasmus/",
"example.com", 1);
?>
```

```
<?php
//Doar pe urmatoarele pagini !!!!

// Print an individual cookie
echo $_COOKIE["TestCookie"];

// Another way to debug/test is to view all cookies
print_r($_COOKIE);
?>
```

Cookies

```
<?php|
//Cookie arrays
// set the cookies
setcookie("cookie[three]", "cookiethree");
setcookie("cookie[two]", "cookietwo");
setcookie("cookie[one]", "cookieone");

// after the page reloads, print them out
if (isset($_COOKIE['cookie']))
{
    foreach ($_COOKIE['cookie'] as $name => $value)
    {
        $name = htmlspecialchars($name);
        $value = htmlspecialchars($value);
        echo "$name : $value <br />\n";
    }
}

?>
```

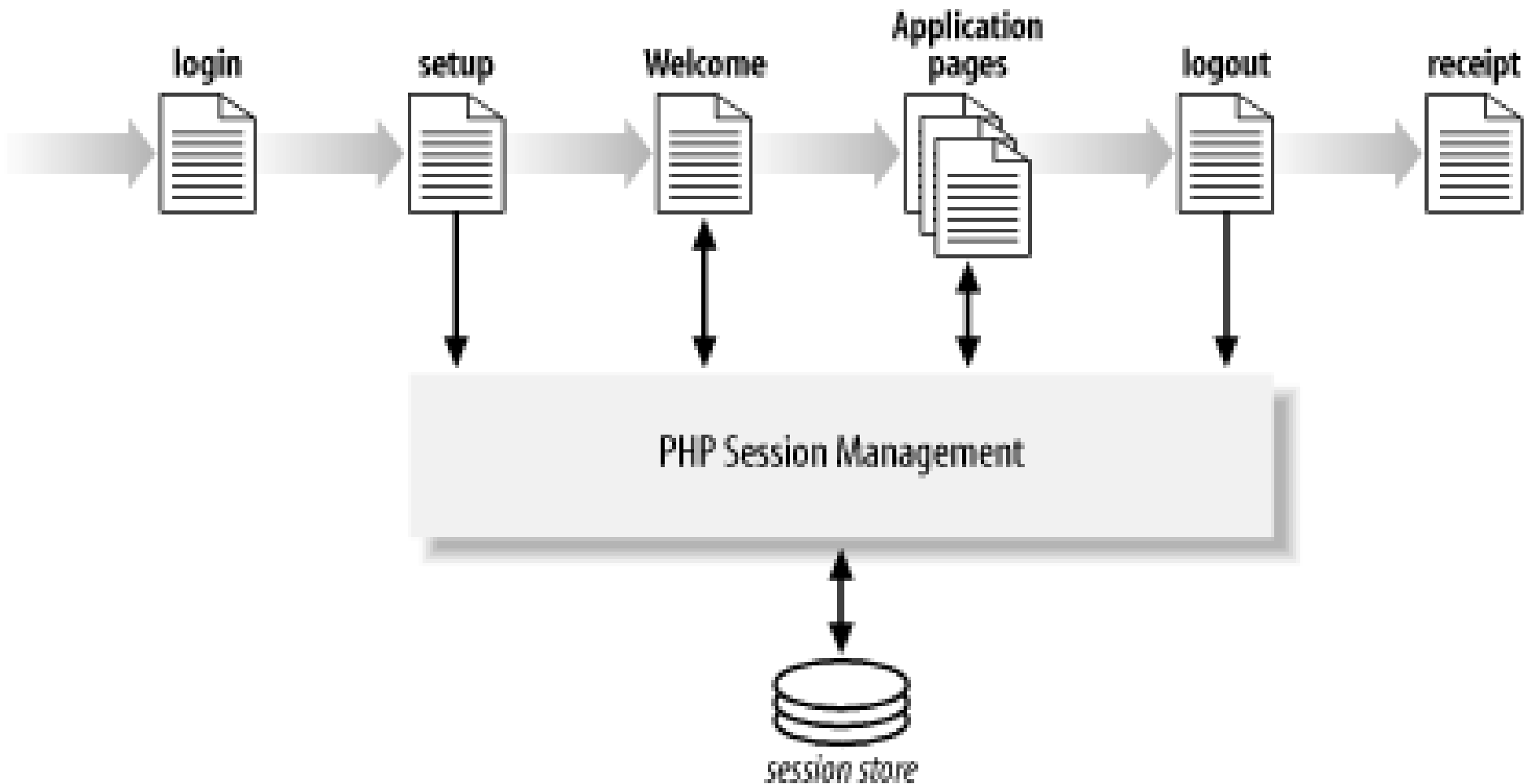
Sesiune

- cookie poate oferi "memorie" aplicatiilor web
- dezavantaje
 - datele se stocheaza la client, nu sunt in siguranta
 - nu se pot stoca oricate date (max. 20)
 - e posibil clientul sa nu accepte cookie
- Sesiunea pentru evitarea acestor dezavantaje
 - stocare pe server
 - oricat de mult date
 - daca clientul nu accepta cookie, "memoria" se realizeaza prin metoda "get"

Sesiune

- `session_start();` (session_ID din GET, POST, COOKIE)
- `session_write_close ();`
- `session_id ([string id]);`
- datele se manipuleaza prin variabila globala `$_SESSION` care ofera acces la citirea/scrierea datelor

Sesiune



Sesiune

SESSIONS Management



Sesiune

```
<?php
// Initialize the session.
// If you are using session_name("something"), don't forget it now!
session_start();

// Unset all of the session variables.
$_SESSION = array();

// If it's desired to kill the session, also delete the session cookie.
// Note: This will destroy the session, and not just the session data!
if (isset($_COOKIE[session_name()]))
{
    setcookie(session_name(), '', time()-42000, '/');
}

// Finally, destroy the session.
session_destroy();?>
```


Sesiune

```
<?php
// page1.php

session_start();

echo 'Welcome to page #1';

$_SESSION['favcolor'] = 'green';
$_SESSION['animal'] = 'cat';
$_SESSION['time'] = time();

// Works if session cookie was accepted
echo '<br /><a href="page2.php">page 2</a>';

// Or maybe pass along the session id, if needed
//echo '<br /><a href="page2.php?" . SID . ">page 2</a>';
echo '<a href="page2.php?" . session_name() . " = ' .
session_id() . "'>page2</a>' ;
?>
```

Sesiune

```
<?php|
// page2.php

session_start();

echo 'Welcome to page #2<br />';

echo $_SESSION['favcolor']; // green
echo $_SESSION['animal'];   // cat
echo date('Y m d H:i:s', $_SESSION['time']);

// You may want to use SID here, like we did in page1.php
echo '<br /><a href="page1.php">page 1</a>';
?>
```

Contact

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