Lecture 11 (w12)

2025/2026

Databases, Web Programming and Interfacing

DWPI

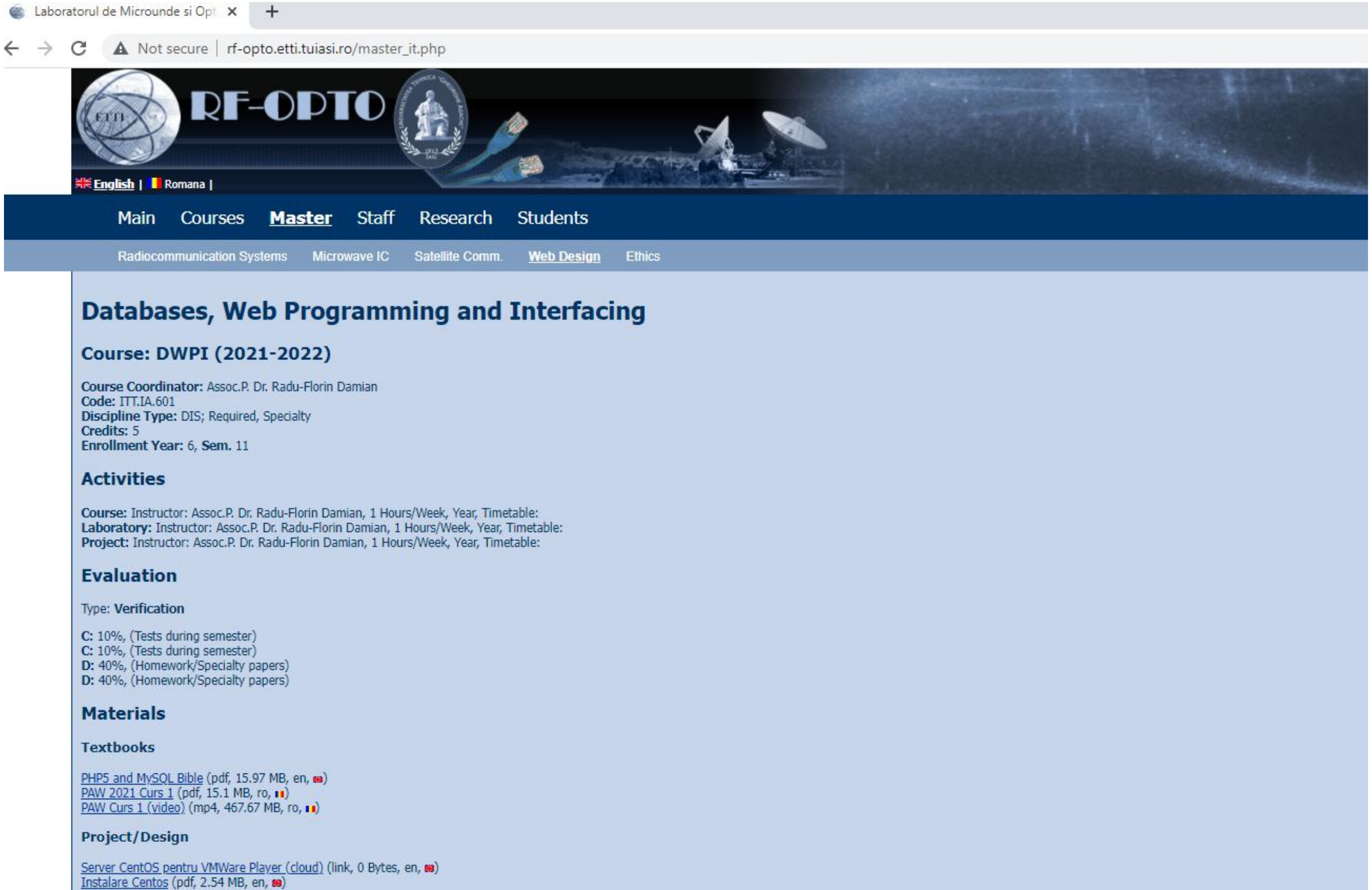
- Databases, Web Programming and Interfacing
  - An VI IT4T
    - 1C/1L/1P
- Timetable
  - Friday, every week (fiecare saptamana) 1C + 2L (17-20)
  - Test + Project Presentation: **last week** (Weekend)

Grade

- 10% - Test/Exam – last week – 1h
- 40% - Project
  - Personal(80%)/Team(20%)
- Previous database project
  - receives a **web user interface**
  - **split** in individual assignments
  - bonus (individual) for **supplemental embedded systems access!**
    - admin/user – web user interface
    - data supplier – embedded system (even partial)

Info

- https://rf-opto.etti.tuiasi.ro/master_it.php



Databases, Web Programming and Interfacing

Course: DWPI (2021-2022)

Course Coordinator: Assoc.P. Dr. Radu-Florin Damian
Code: ITT.IA.601
Discipline Type: DIS; Required, Specialty
Credits: 5
Enrollment Year: 6, **Sem.** 11

Activities

Course: Instructor: Assoc.P. Dr. Radu-Florin Damian, 1 Hours/Week, Year, Timetable:
Laboratory: Instructor: Assoc.P. Dr. Radu-Florin Damian, 1 Hours/Week, Year, Timetable:
Project: Instructor: Assoc.P. Dr. Radu-Florin Damian, 1 Hours/Week, Year, Timetable:

Evaluation

Type: **Verification**

C: 10%, (Tests during semester)
C: 10%, (Tests during semester)
D: 40%, (Homework/Specialty papers)
D: 40%, (Homework/Specialty papers)

Materials

Textbooks

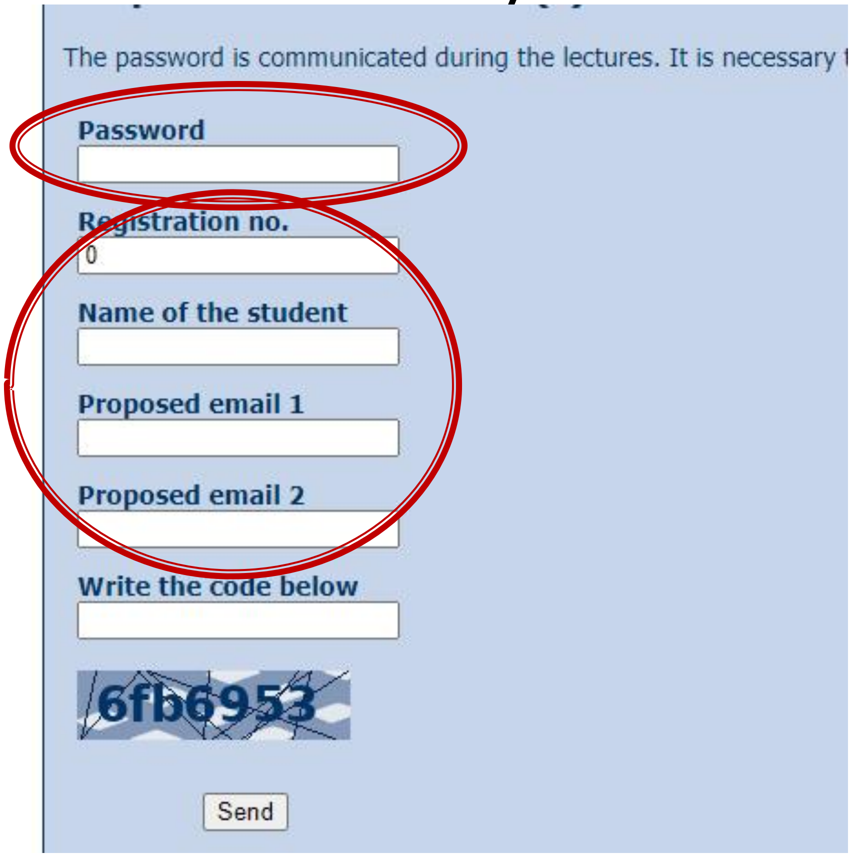
PHP5 and MySQL Bible (pdf, 15.97 MB, en, 🇬🇧)
PAW 2021 Curs 1 (pdf, 15.1 MB, ro, 🇷🇴)
PAW Curs 1 (video) (mp4, 467.67 MB, ro, 🇷🇴)

Project/Design

Server CentOS pentru VMWare Player (cloud) (link, 0 Bytes, en, 🇬🇧)
Instalare Centos (pdf, 2.54 MB, en, 🇬🇧)

Online – Registration no.

- access to **online exams** requires the **password** received by email



The password is communicated during the lectures. It is necessary t

Password

Registration no.
0

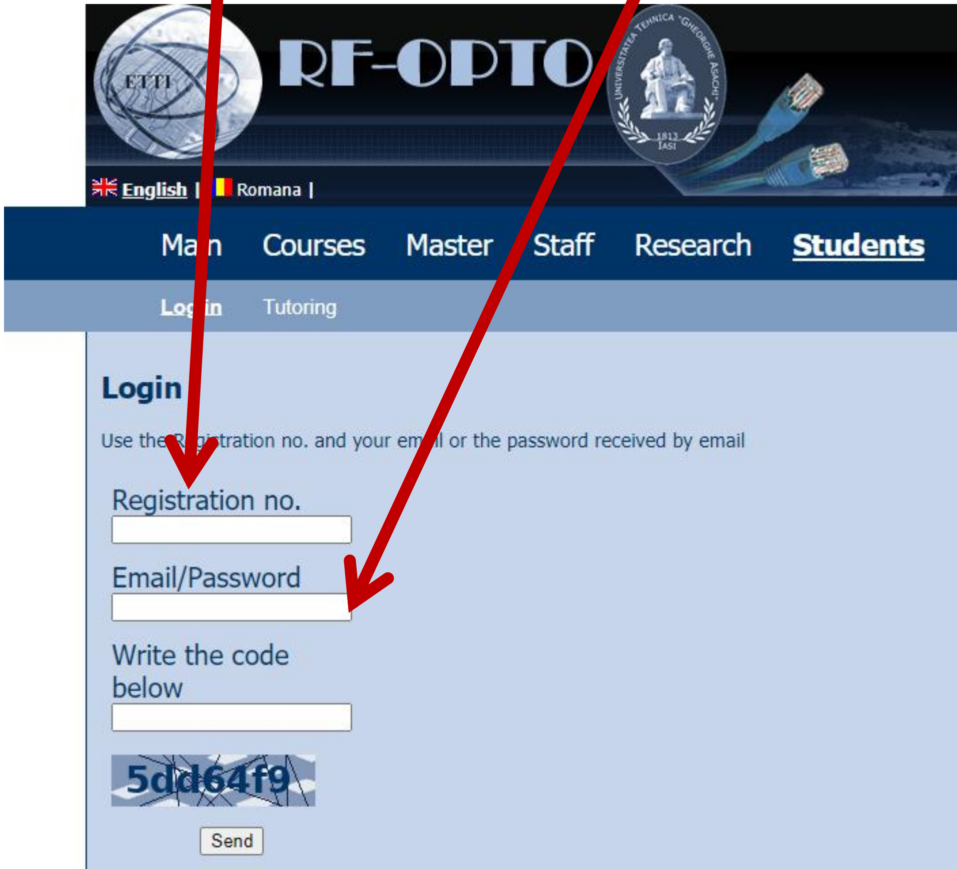
Name of the student

Proposed email 1

Proposed email 2

Write the code below

6fb6953



Login

Use the Registration no. and your email or the password received by email

Registration no.

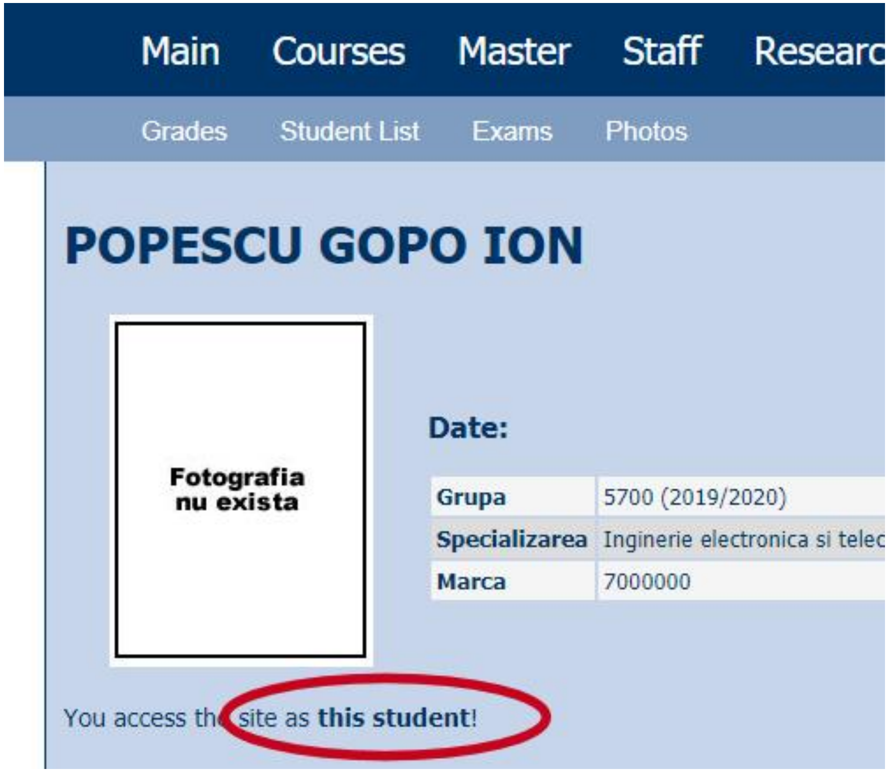
Email/Password

Write the code below

5dd64f9

Online

- access email/password

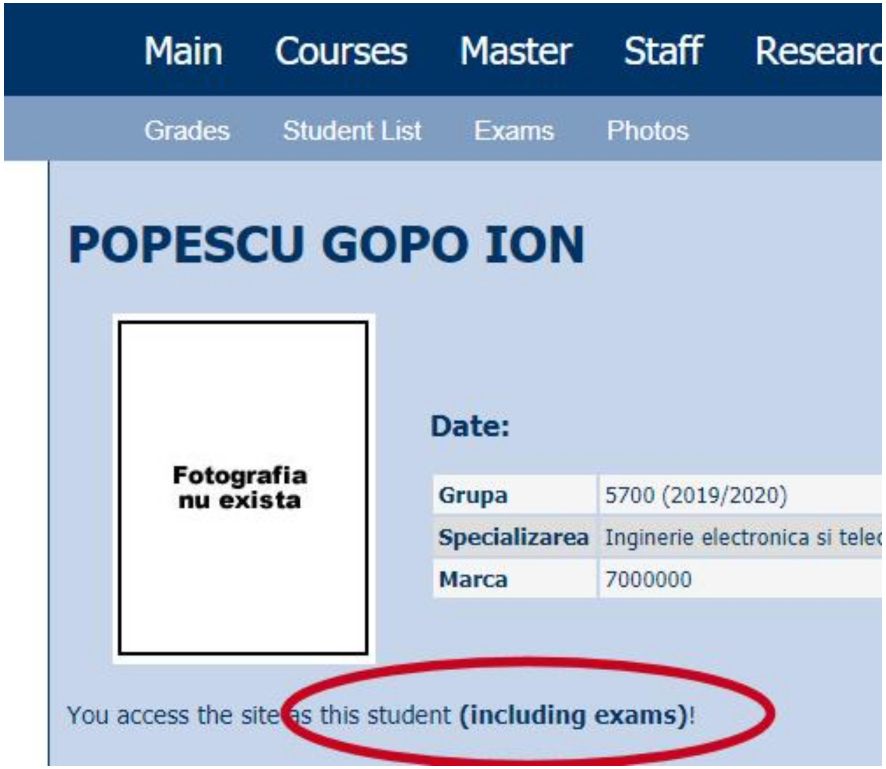


POPESCU GOPO ION

Date:

Grupa	5700 (2019/2020)
Specializarea	Inginerie electronica si telec
Marca	7000000

You access the site as **this student**!



POPESCU GOPO ION

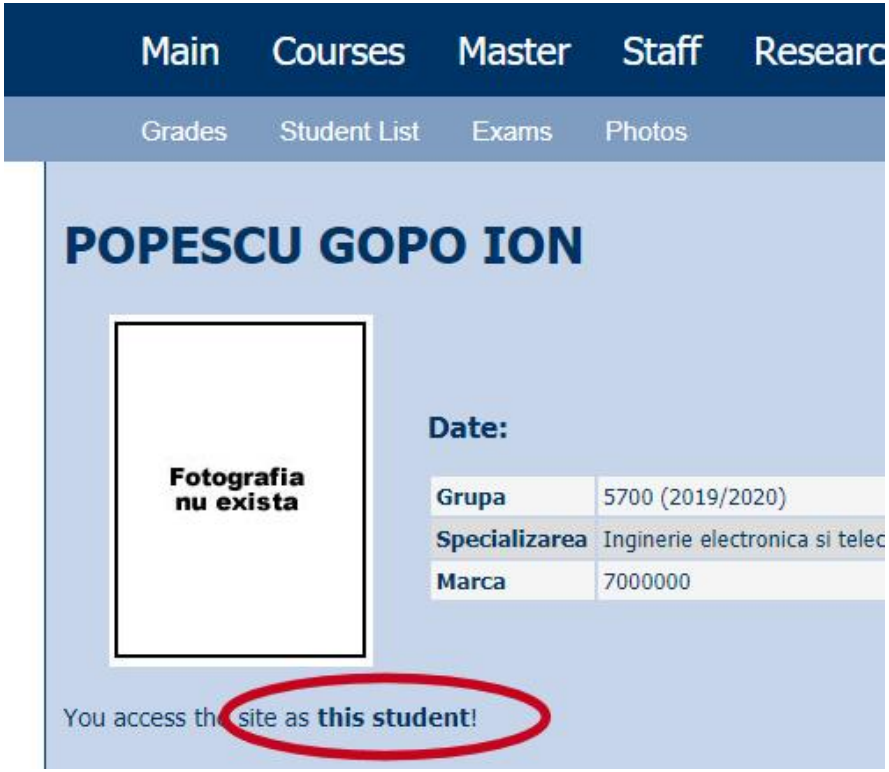
Date:

Grupa	5700 (2019/2020)
Specializarea	Inginerie electronica si telec
Marca	7000000

You access the site as this student **(including exams)**!

Online

- access email/password



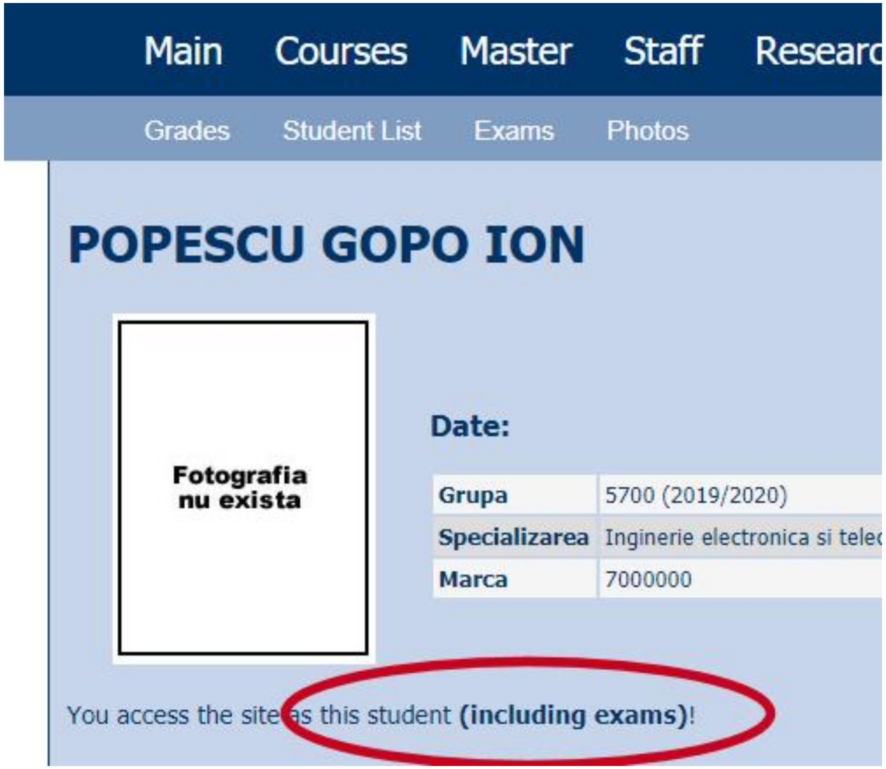
POPESCU GOPO ION

Fotografia nu exista

Date:

Grupa	5700 (2019/2020)
Specializarea	Inginerie electronica si telec
Marca	7000000

You access the site as **this student**!



POPESCU GOPO ION

Fotografia nu exista

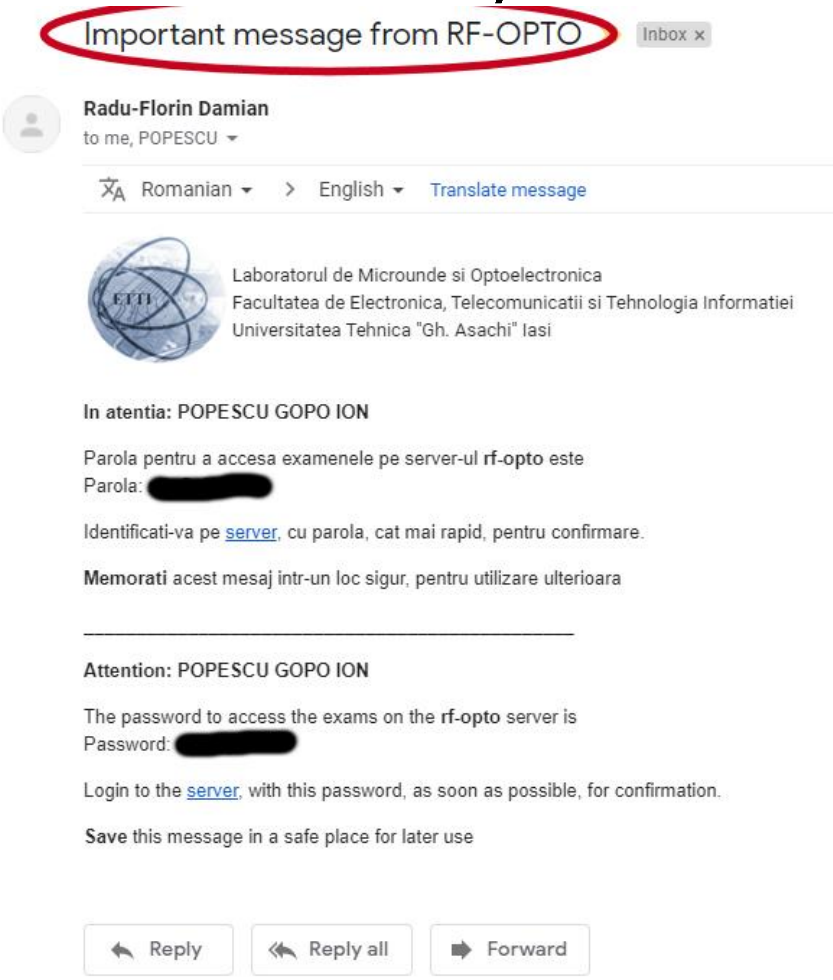
Date:

Grupa	5700 (2019/2020)
Specializarea	Inginerie electronica si telec
Marca	7000000

You access the site as this student **(including exams)**!

Password

- received by email



Important message from RF-OPTO

Radu-Florin Damian

to me, POPESCU

Laboratorul de Microunde si Optoelectronica
Facultatea de Electronica, Telecomunicatii si Tehnologia Informatiei
Universitatea Tehnica "Gh. Asachi" Iasi

In atentia: POPESCU GOPO ION

Parola pentru a accesa examenele pe server-ul **rf-opto** este
Parola: [redacted]

Identificati-va pe server, cu parola, cat mai rapid, pentru confirmare.

Memorati acest mesaj intr-un loc sigur, pentru utilizare ulterioara

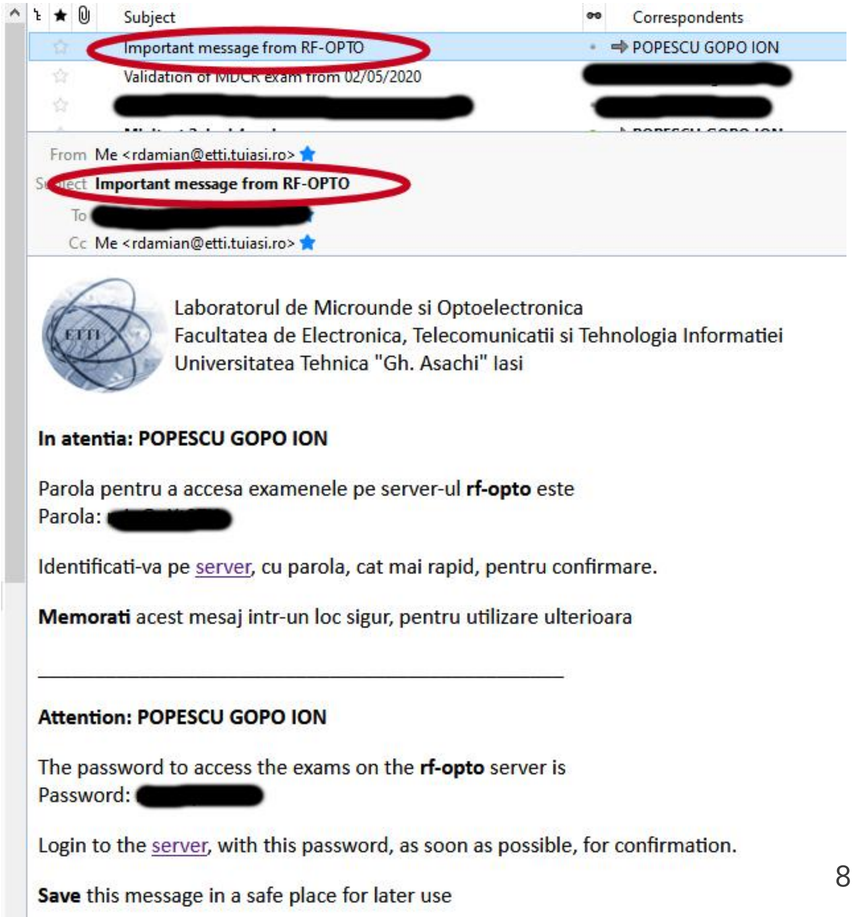
____

Attention: POPESCU GOPO ION

The password to access the exams on the **rf-opto** server is
Password: [redacted]

Login to the server, with this password, as soon as possible, for confirmation.

Save this message in a safe place for later use



Subject	Correspondents
Important message from RF-OPTO	POPESCU GOPO ION
Validation of MDCR exam from 02/05/2020	[redacted]
[redacted]	[redacted]

From Me <rdamian@etti.tuiasi.ro>

Subject **Important message from RF-OPTO**

To [redacted]

Cc Me <rdamian@etti.tuiasi.ro>

Laboratorul de Microunde si Optoelectronica
Facultatea de Electronica, Telecomunicatii si Tehnologia Informatiei
Universitatea Tehnica "Gh. Asachi" Iasi

In atentia: POPESCU GOPO ION

Parola pentru a accesa examenele pe server-ul **rf-opto** este
Parola: [redacted]

Identificati-va pe server, cu parola, cat mai rapid, pentru confirmare.

Memorati acest mesaj intr-un loc sigur, pentru utilizare ulterioara

____

Attention: POPESCU GOPO ION

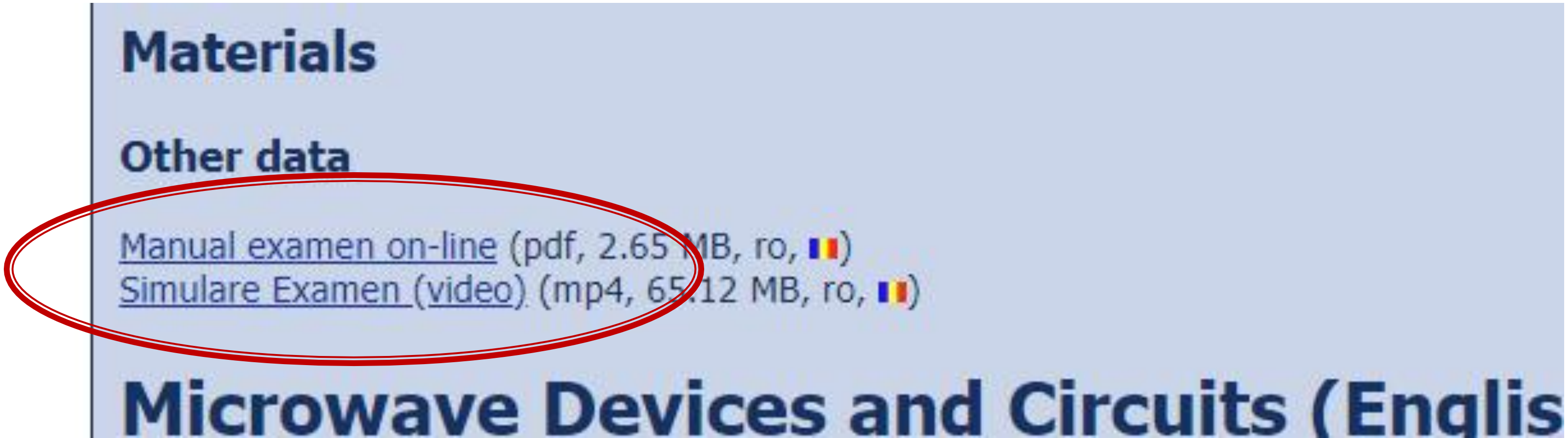
The password to access the exams on the **rf-opto** server is
Password: [redacted]

Login to the server, with this password, as soon as possible, for confirmation.

Save this message in a safe place for later use

Manual examen online

- The online exam app used for:
  - lectures (attendance)
  - laboratory
  - project
  - examinations



Materials

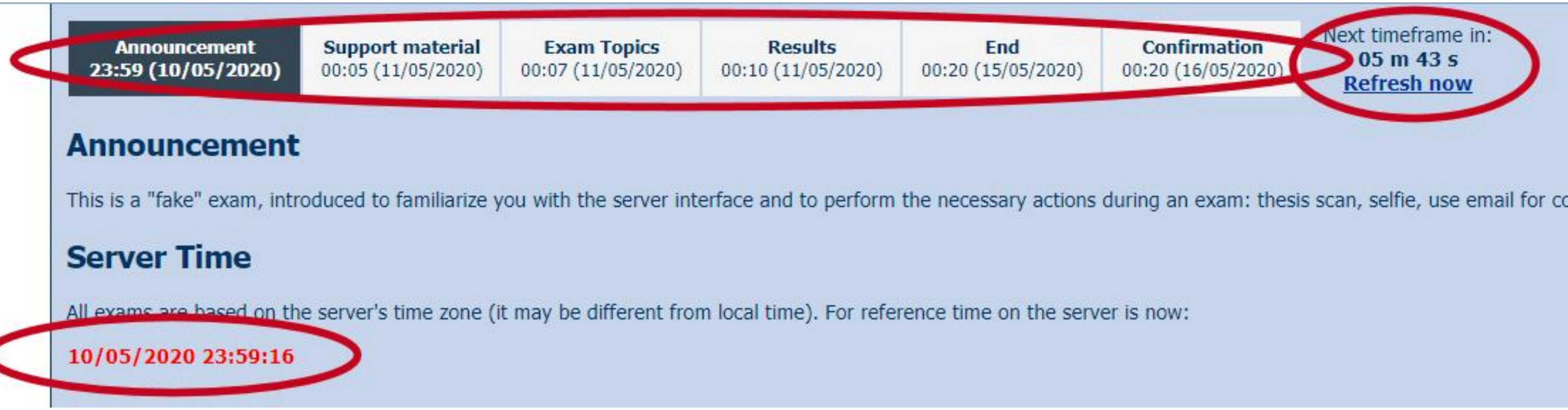
Other data

Manual examen on-line (pdf, 2.65 MB, ro, 🇷🇴)
Simulare Examen (video) (mp4, 65.12 MB, ro, 🇷🇴)

Microwave Devices and Circuits (Englis

Examen online

- always against a **timetable**
  - long period (project submission/laboratory results)
  - short period (tests: 15min, exam: 2h)



Announcement	Support material	Exam Topics	Results	End	Confirmation
23:59 (10/05/2020)	00:05 (11/05/2020)	00:07 (11/05/2020)	00:10 (11/05/2020)	00:20 (15/05/2020)	00:20 (16/05/2020)

Next timeframe in:
05 m 43 s
Refresh now

Announcement

This is a "fake" exam, introduced to familiarize you with the server interface and to perform the necessary actions during an exam: thesis scan, selfie, use email for co

Server Time

All exams are based on the server's time zone (it may be different from local time). For reference time on the server is now:

10/05/2020 23:59:16

Email addresses

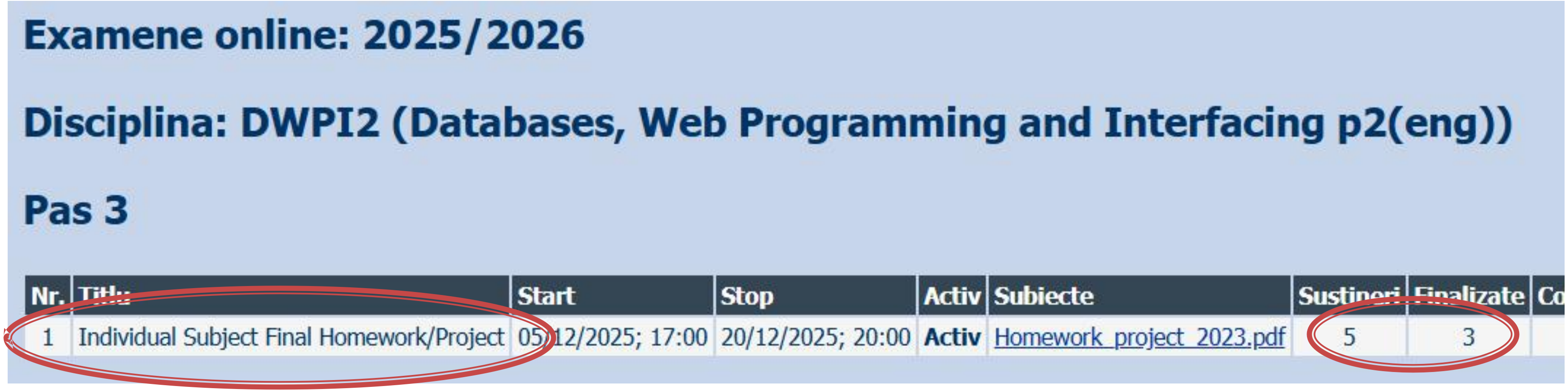
- Team representative
  - list of **active** email addresses for all students
    - can be @student.etti.tuiasi.ro (also @gmail ~~@yahoo~~ etc.)
  - **rdamian@etti.tuiasi.ro**

2025/2026

Project

Project

- self selected homework: available until **20.12** (~**1 day!!**)
- **only** selection from a list after 20.12



Examene online: 2025/2026

Disciplina: DWPI2 (Databases, Web Programming and Interfacing p2(eng))

Pas 3

Nr.	Titlu	Start	Stop	Activ	Subiecte	Sustineri	Finalizate	Co
1	Individual Subject Final Homework/Project	05/12/2025; 17:00	20/12/2025; 20:00	**Activ**	Homework_project_2023.pdf	5	3	

Project

- Submission: **On-site**
- Presentation (in front of the colleagues) + files submission
- 3 files
  - **1 *.pdf** (print-screen while using the application, short usage instructions, a mini-user manual for the application) (**personal**)
  - **1 *.sql** (backup of the database required to run the application) (**team**)
  - archive of the application (inside: files *.php, *.jpg, folder tree etc., archived: ***.zip, *.7z** etc.) (**team/personal**)

Project grading

- **(2p)** the application runs on the **reference server** (can be downloaded from rf-opto: Debian, php 8 **or** Ubuntu, php 7 **or** CentOS 7, php 5): extract files from the ***.zip** archive in a folder on the server, restore database from the ***.sql** backup file
- **(2p)** the ***. pdf** file containing the user manual exists and is appropriate for the submitted application
- **(2p)** the application **flowchart** has been submitted and contains appropriate data
- **(4p)** presentation on-site of the **application**

Online exam 05.12 - 20.12

- on-line exam for **choosing** the **individual** assignment **inside** the **team** (/individual) web **application**
- Data:
  - 1. **Selfie**, **file**, taken during the submission process (acting as a signature for the submission)
  - 2. **Title** of your (team) application/project, **text** (e.g. "Student Enrollment")
  - 3. **Team members**, **text** (e.g. "Popescu/Ionescu/Vasilescu")
  - 4. **Database**, **file**, Access (*.accdb) or SQL (*.sql), current database
  - 5. **Individual assignment**, **file**, Doc/Pdf/Txt describing the proposed individual activity (e.g. "I intend to design the scripts that will: 1. add records in the Students table + 2. extract the full list from Courses table + 3. list records from Students table following specified criteria")

Online exam 05.12 - 20.12

- While it is not required to keep the same team composition and/or the same database from the first half of the semester, it is **recommended**.
- The current database is not required to be kept identical until the end of the semester. Addition of tables/columns or change of data types is to be expected during application development. The current file is meant to be used only as support for understanding the proposed individual assignment.
- The individual assignment must not overlap with another team member assignment and cannot be changed during the semester. However, deviation from the assignment will be permitted (**only** if properly justified during the final presentation)
- A team integrated final application (same design, common application) will beneficiate from a bonus during the final submission. However individual grade will only depend on fulfillment of the individual assignment. Maximum grade can be obtained without the team bonus.

Online exam 19.12 – xx.06

- on-line exam for the submission of the final individual work inside the team (/individual) web application.
- Data:
  - 1. **Selfie**, **file**, taken during the submission process (acting as a signature for the submission)
  - 2. **Title** of your (team) application/project, **text**
  - 3. **Team members**, **text**
  - 4. **PDF file, file**, includes a **flowchart** of your team application with identification of **individual assignment** inside the team application, and print-screen while using the application, short usage instructions, a mini-user manual for the application (**only** for your **individual assignment**)
  - 4. **SQL file**, **file**, (*.sql), backup of the final database required to run your submitted application
  - 5. **Archive file**, **file**, one archive of the application (inside: files *.php, *.jpg, folder tree etc., archived: *.zip, *.7z etc.)

Online exam 19.12 – xx.06

- The files must be uploaded by **all team members, even if they are identical** (access to the common files is a minimal proof of team membership).
- The "SQL file" and "Archive file" can (should) be common, as is the case for the overall flowchart of the team application.
- However, the identification of individual assignment on the flowchart and the mini-user manual (print-screen, usage instructions) **must** be different between team members, without overlaps.

Hypertext PreProcessor

PHP

Escaping from HTML

- A PHP file normally contains HTML tags (**IS** basically a HTML file), with some PHP code **sections** inside
- PHP interpreter
- the PHP interpreter looks for sections that it needs to interpret and their interior is processed as PHP code
- what is found **outside** these sections is sent to the web server **unmodified**
- most of the time the result is a virtual "text file" containing (pure) **HTML** code (+CSS/Javascript)

Escaping from HTML

- **<?php … ?>**
  - XML type – default, always available, recommended
- <? … ?>
  - short, usually deactivated
  - absent in PHP 7,8
- <script language=“php”> … </script>
  - script type, available
- <% … %>
  - ASP type, usually deactivated
  - absent in PHP 7,8

Escaping from HTML

- **echo** …. a PHP language construct: output one or more strings (equivalent with puts() in C)
- can process data
  - echo $a + $b;
- in most cases “output” is the data sent to the client by the web server
- “output” can usually be considered:
  - the current document
  - that particular script position inside the document

Escaping possibilities

- All versions offer the same HTML source for the browser
- The one that leaves the HTML structure outside the escaping tags and only the dynamic data results from PHP processing is **recommended**
- HTML + PHP code is interpreted much more elegantly in WYSIWYG editors

```
<h2>Transaction result</h2>
<?php echo '<p>Command received</p>';?>
```

```
<h2>Transaction result</h2>
<p><?php echo 'Command received';?></p>
```

```
<?php echo '<h1>Online Shop XXX SRL</h1>';?>
<?php echo '<h2>Transaction result</h2>';?>
<?php echo '<p>Command received</p>';?>
```

```
<?php
echo '<h1>Online Shop XXX SRL</h1>‘;
echo '<h2>Transaction result</h2>';
echo '<p>Command received</p>';
?>
```

PHP – variables

- variable – **$** character followed by the variable name
- the variable name is “case sensitive”
- a frequent mistake is forgetting the $ char
  - PHP Notice: Use of undefined constant an – assumed $an (or ‘an’) in D:\\Server\\
- Data types
  - scalar
  - composite
  - special

PHP – data types

- scalar
  - Boolean
  - integer
  - float (double)
  - **string**
- composite
  - array
  - object
- special
  - resource
  - NULL

PHP – Variables

- declaring variables is **not required**, except when declaring a definition domain (eg. global variables)
  - global $a, $b;
    $c=$a+$b;
- freeing the memory is not necessary, PHP does it automatically at the end of the script

PHP – Variables

$var = **expression**

- Control of variables is automatic, “on the fly”
  - A variable starts with the **$** sign, followed by the name of the variable
  - PHP has no command for declaring a variable. It is created the moment you first assign a value to it
  - PHP automatically associates a data type to the variable, depending on its value
  - Variable names are case-sensitive
    ($age and $AGE are two different variables)

PHP – data types

- PHP automatically associates a data type to the variable, depending on its value
- automatic conversions are often **not** numerical but "**human**"
- In PHP 7, type declarations were added for function arguments, return values and class properties

```
<?php
echo $variable ;  //  type Null, not initialized – value NULL (only)
$variable  = "0";  // $ variable   type  string (ASCII 48)
$variable  += 2;   // $ variable   type  integer (2)
$variable  = $ variable  + 1.3;  // $variable   type  float (3.3)
$variable  = 5 + "10 obiects"; // $variable   type  integer (15)
$var2=5; // $var2   type  integer (5)
$variable =$var2."10 objects";    //  $variable   type  string  "510 objects"
?>
```

PHP – operators

- Similar (generally) to C/C++
- An operator takes one or more values (or expressions), performs an operation and yields another value
- Operators are divided in three groups
  - Unary operators: applied to one value
  - Binary operators: applied to two values
  - Ternary operators: applied to three values

PHP – operators

- Operators
  - Arithmetic
  - Assignment
  - Bitwise
  - Comparison
  - Incrementing/Decrementing
  - Logical
  - **String**

PHP – operators

- Arithmetic
  - -$a – Negation
  - $a + $b – Addition
  - $a - $b – Subtraction
  - $a * $b – Multiplication
  - $a / $b – Division
  - $a % $b – Modulo (remainder)
  - $a ** $b – Exponentiation
- **String**
  - **$a.$b – Concatenation String a with String b**

PHP – operators

- Assignment
  - $a=$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^b)
  - **$a.=$b (a=a concatenate b - strings)**

Control Structures

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

- includes **and** evaluates the specified file
- used to avoid multiple evaluations of common code sections
- require will halt the script if the specified file is not found
- ..._once() checks if the code from a file has already been included and it will **not** be included **again**

String Variables

String Variables ‘ ’

- single quotes **‘ ’** are used for the definition of classic basic strings
  - a set of characters is defined
  - processing inside the string is reduced
    - \’ is a literal single quote
    - \\ and \ are a literal backslash
    - **only!!!**

String Variables “ ”

- double-quotes **“ ”** are used for the definition of complex strings
  - complex processing inside the string more than the C/C++ equivalent
    - special ASCII characters, similar to C++: \n, \r, \t, \\, \v, \e, \f, \x, \u
    - \” character double-quote
    - \$ character $
    - **variable names** inside the string will be expanded !!!

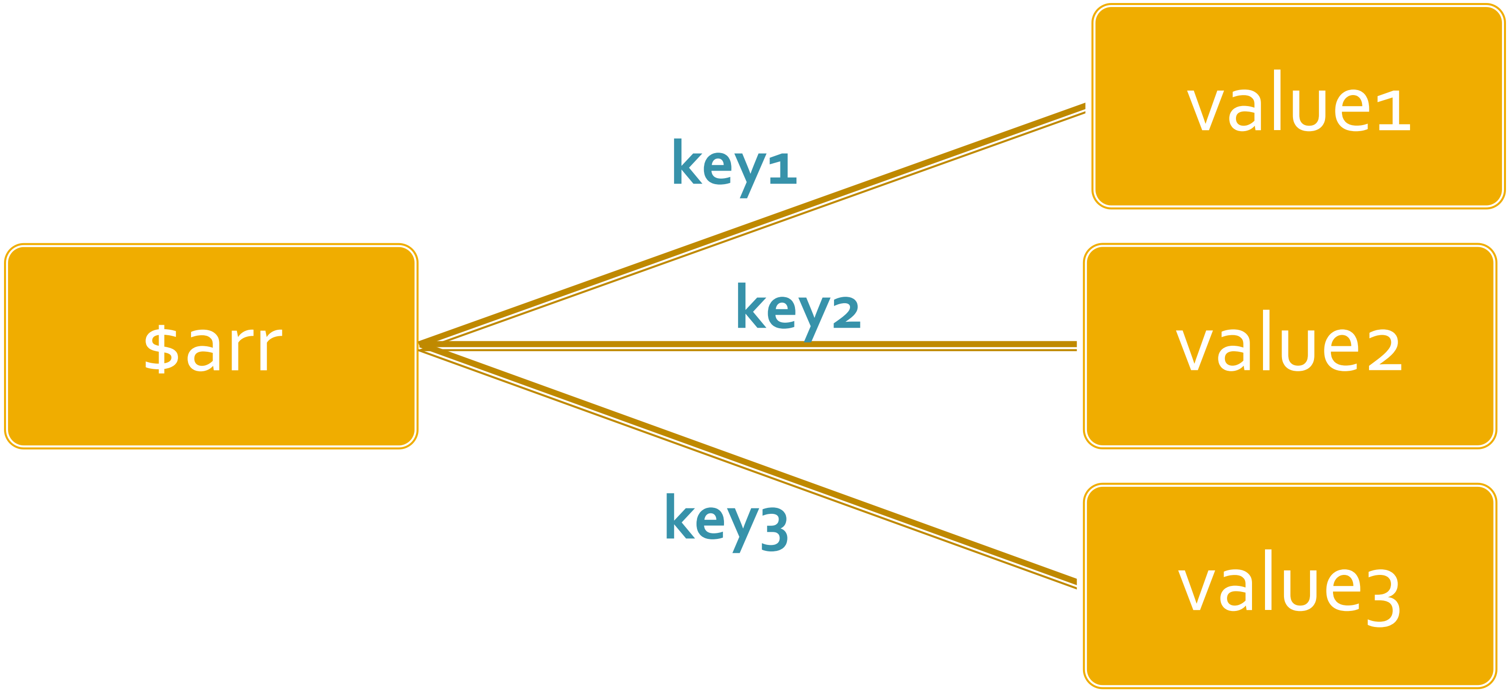
Arrays

Arrays in PHP

- An array in PHP is actually an ordered map. A map is a type that associates **values** to **keys**
- unlike C, Basic, **keys** are **not** required to be **integers**, can be **strings**
- default keys (if not otherwise specified) are consecutive integers with first key 0 (C syntax).
- defining a key / value pair
  - key **=>** value
- create an array
  - $arr = **array**("definition of key / value pairs")
  - pairs: key **=>** value, key **=>** value, ...

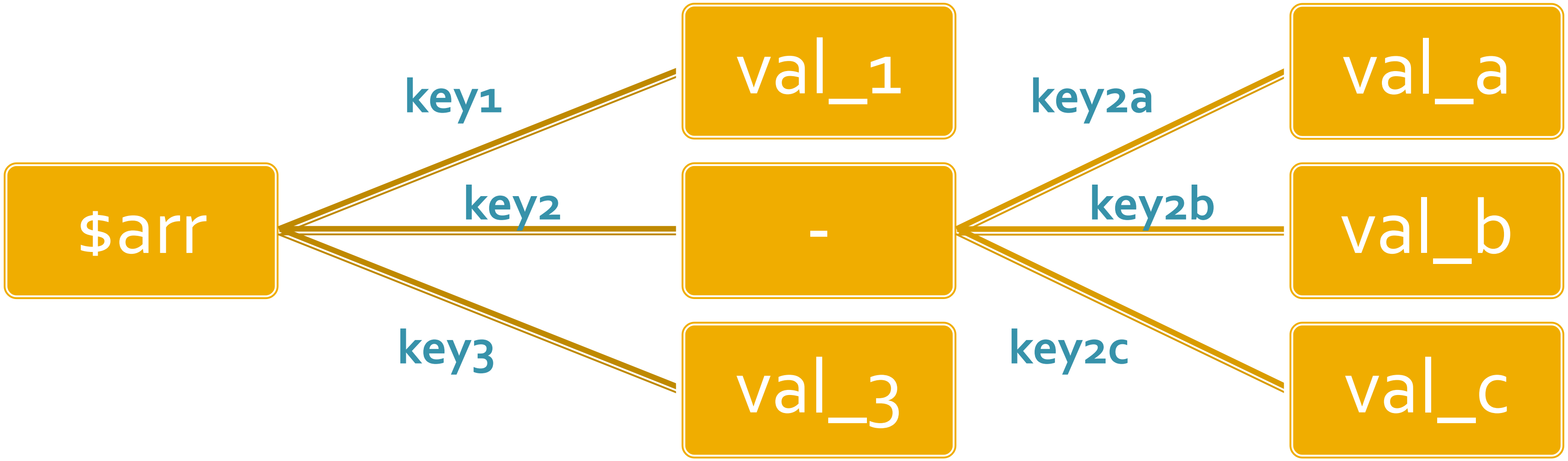
Array = Logical tree

- $arr = array(key1 => value1, key2 => value2, key3 => value3)

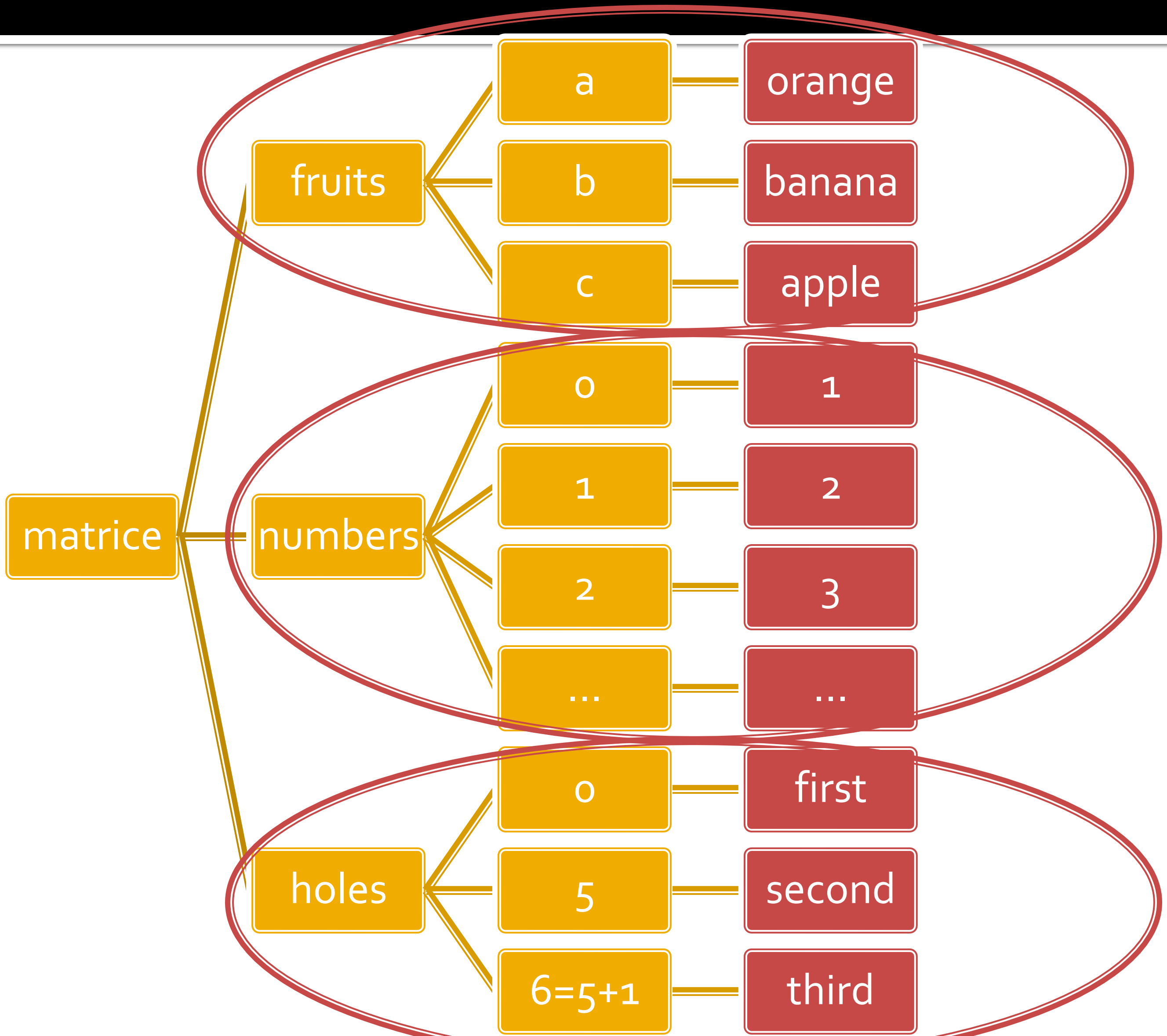


Array = Logical tree

- In particular, one or more of the values can in turn be an array, leading to **branching** of the tree
- $arr = array(key1 **=>** val_1, **key2 => array**(key2a **=>** val_a, key2b **=>** val_b, key2c **=>** val_c), key3 **=>** val_3)



Arrays in PHP



matrice

fruits

a orange

b banana

c apple

numbers

0 1

1 2

2 3

... ...

holes

0 first

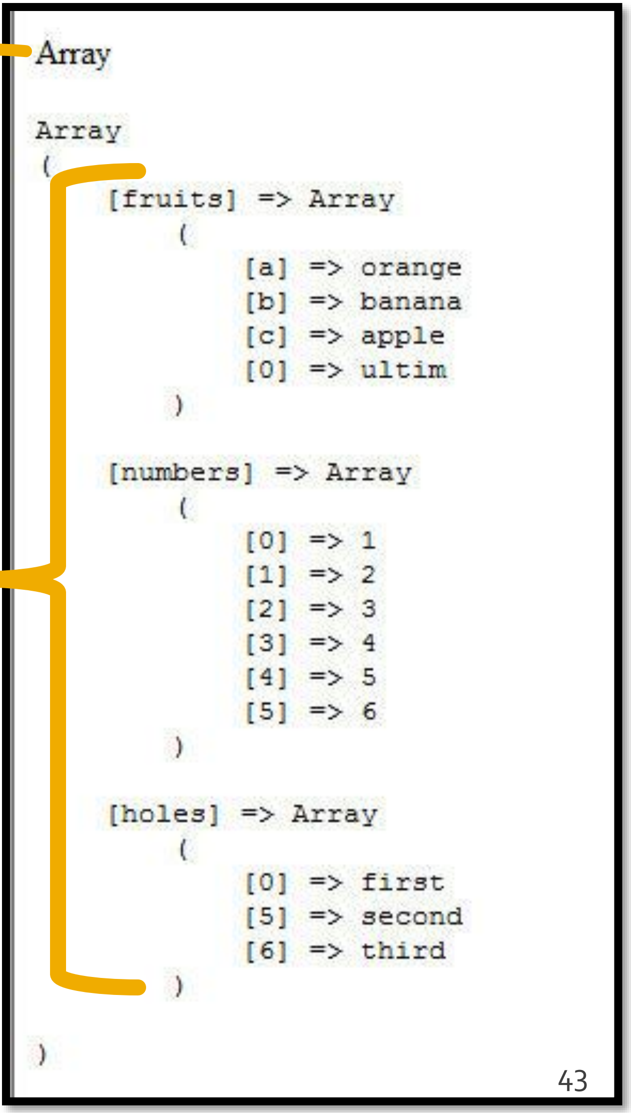
5 second

6=5+1 third

View array content (debug)

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

```
$matr= array (
"fruits"  =>
array("a" => "orange", "b" => "banana", "c" => "apple" ,
"ultim"),
"numbers" =>
array(1, 2, 3, 4, 5, 6),
"holes"   =>
array("first", 5 => "second", "third")
);
echo $matr;
echo "<pre>";
print_r ($matr);
echo "</pre>";
```



Array

```
Array
(
    [fruits] => Array
        (
            [a] => orange
            [b] => banana
            [c] => apple
            [0] => ultim
        )

    [numbers] => Array
        (
            [0] => 1
            [1] => 2
            [2] => 3
            [3] => 4
            [4] => 5
            [5] => 6
        )

    [holes] => Array
        (
            [0] => first
            [5] => second
            [6] => third
        )

)
```

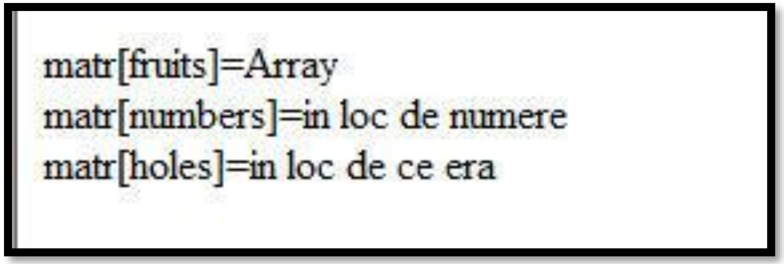
Foreach loop

- **foreach (array_expression as $key => $value)**
  statement
- **foreach (array_expression as $value) statement**
- foreach construct is used to loop through each key/value pair in an array
- On each iteration assign the current element's key to the local variable **$key** and the value of the current element is assigned to the local variable **$value** (scope: statement)
- foreach() works with a **copy** of the array, you cannot change the original array inside the statement

  -
    ```
    foreach ($arr as $key => $value) {
    $value = 'other value'; //doesn't work
    $arr[$key] = 'other value'; //works
    }
    ```

Example – foreach

```
$matr = array (
   "fruits"  => array("a" => "orange", "b" => "banana", "c" => "apple", "ultim"),
   "numbers" => "in loc de numere",
   "holes"   => "in loc de ce era"
);
foreach ($matr as $cheie => $continut)
                  echo "matr[".$cheie."]=".$continut."<br />";
```



```
matr[fruits]=Array
matr[numbers]=in loc de numere
matr[holes]=in loc de ce era
```

PHP Global Variables - Superglobals

PHP Global Variables - Superglobals

- PHP Global Variables - Superglobals (predefined variables)
  - are always accessible, regardless of scope
  - Examples:
    - **$_SERVER** — Server and execution environment information
    - **$_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

Getting user submitted data

- When a user submits the data by clicking on "Submit", the form data is found in the file specified in the **action** attribute of the <form> tag in one of the superglobal variables:
  - $_POST – method=“post”
  - $_GET – method=“get”
  - $_REQUEST – both methods
- the superglobal variables are **arrays** with **string keys** controlled by the **name** attribute of the input element
  - <input type="text" name="**books_quant**" size="3" maxlength="3" />
  - $_POST[‘**books_quant**’] contains the user input in the receiving script

Organizing $_POST

- **name** attributes in the form inputs become **keys** in the superglobal array $_POST
  - <input type="text" name="**books_quant**" size="3" maxlength="3" />
  - $_POST[‘**books_quant**’] contains the user input
- creating **name** "array like", we can control branching of $_POST grouping input elements in the form as required
  - <input type="text“ name="**quant[books]**" size="3" maxlength="3" />
  - $_POST [‘**quant**’] [‘**books**’] contains the user input

Accessing a MySQL server from PHP

Accessing a MySQL server from PHP

- PHP has two extensions in order to interact with a MySql server (local or remote), these must be activated in php.ini.
  - mysql
  - mysqli (improved: functions for MySql > 4.1)
- A database can be accessed if the PHP script knows a MySql server user with access rights
  - usually every application has its specific MySql user with specific access rights
- A database can also be created from PHP, but it is not the recommended method if it is not necessary
  - the code is difficult to implement and used only once

Accessing a MySQL server from PHP

- mysql_connect
  - Open a connection to a MySQL Server
  - resource **mysql_connect** (string server , string user, string password)
  - returns a MySQL link identifier on success or false on failure
- mysql_query
  - Send a MySQL query
  - resource **mysql_query** (string query [, resource link_identifier])
  - result
    - SELECT, SHOW, DESCRIBE or EXPLAIN: returns a resource or false
    - UPDATE, DELETE, DROP, etc: returns true/false

Accessing a MySQL server from PHP

- mysql_fetch_assoc
  - Returns an **associative array** that corresponds to the fetched row and moves the internal data pointer ahead, or **false** if there are no more rows. The **string** keys of the array are the field names (columns) in the DB table
  - array **mysql_fetch_assoc** (resource result)
- mysql_fetch_row
  - Returns an **numerical** array that corresponds to the fetched row, or false
  - array mysql_fetch_row (resource result)

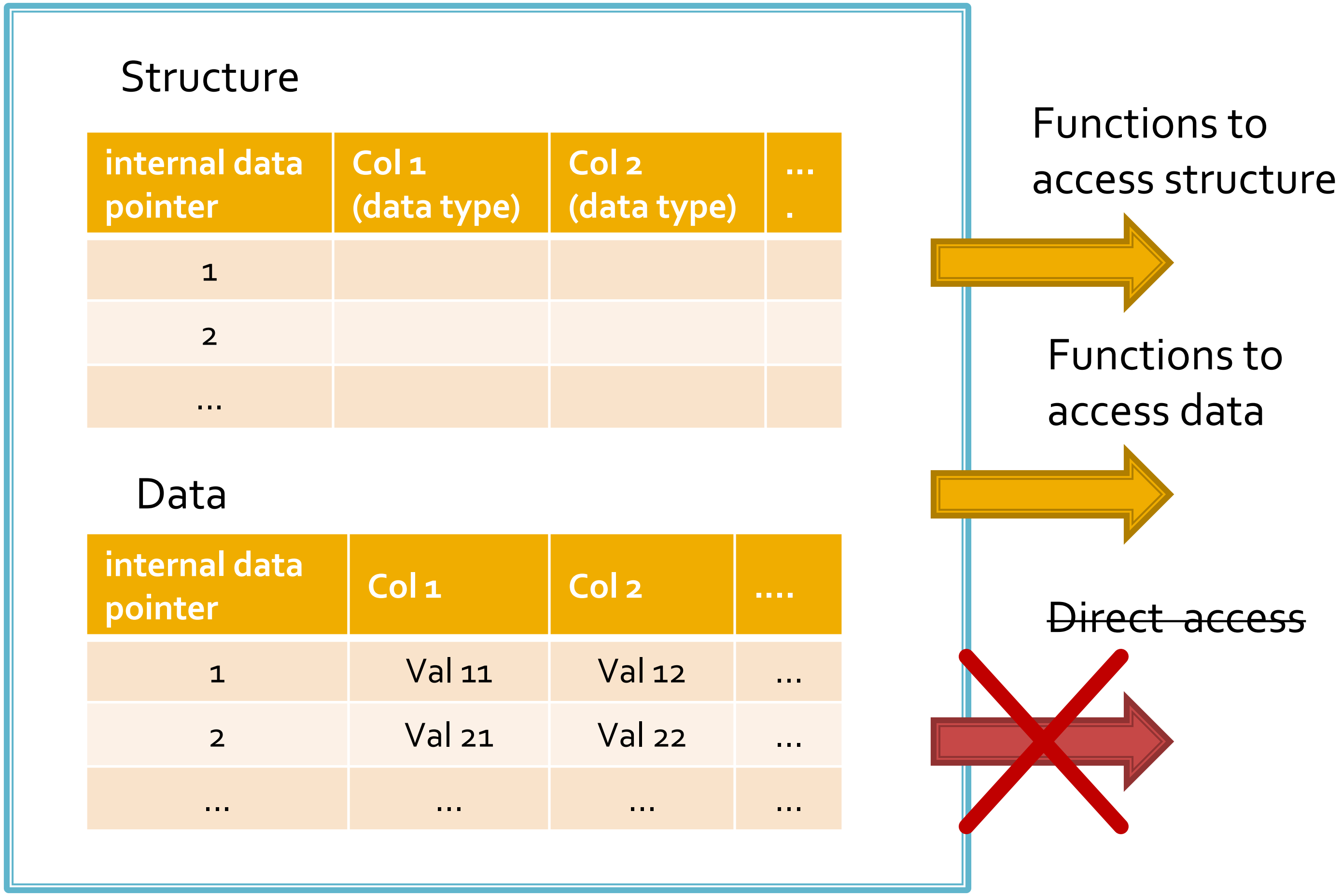
Accessing a MySQL server from PHP

- mysql_fetch_array
  - groups functionality of mysql_fetch_assoc and mysql_fetch_row
  - array **mysql_fetch_array** (resource result [, int result_type])
  - MYSQL_ASSOC, MYSQL_NUM, MYSQL_BOTH (default)
- mysql_data_seek
  - moves the internal row pointer of the MySQL result associated with the specified result identifier to point to the specified row number. The next call to a MySQL fetch function would return that row
  - bool **mysql_data_seek** (resource result, int row_number)

MySQL resources

- Resources are a combination between
  - Structured data (values + structure) resulted from a SQL query
  - functions to access those values/structure
- Analogy with OOP
  - a special "class" created following a SQL query with predefined procedures to access the results of that query

MySQL resources



Structure

internal data pointer	Col 1 (data type)	Col 2 (data type)	….
1			
2			
…			

Data

internal data pointer	Col 1	Col 2	….
1	Val 11	Val 12	…
2	Val 21	Val 22	…
…	…	…	…

Functions to access structure

Functions to access data

~~Direct access~~

MySQL resources

- Structure access functions are rarely used
  - most applications are designed on a fixed DB structure, and the structure of the received data is known
  - exception: general DB applications, eg: PhpMyAdmin
- Most data access functions are characterized by sequential access
  - the data is read line by line
  - simultaneously, internal data pointer advances to the next position, preparing the next read

MySQL resources

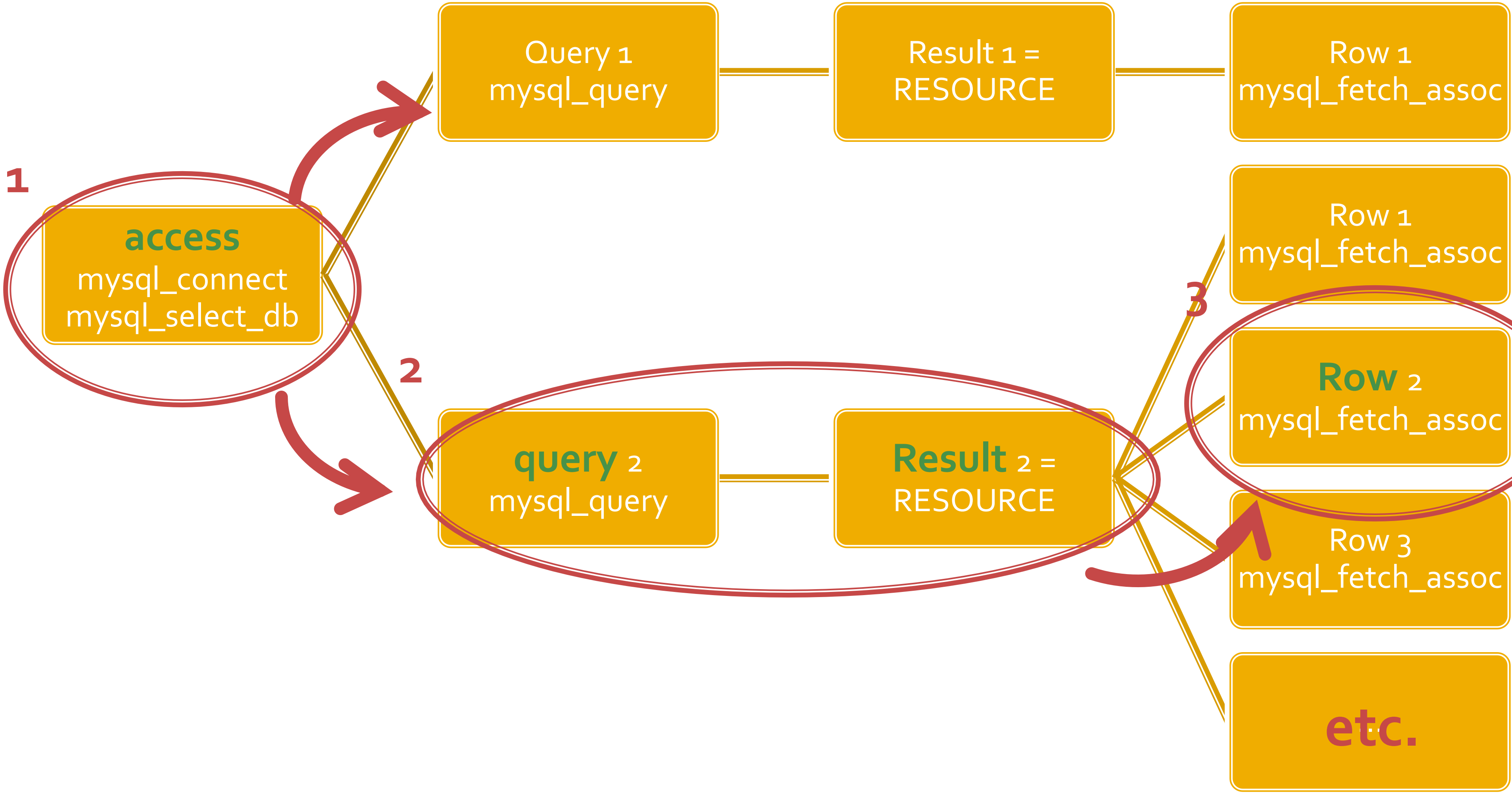
- Functions are optimized for use in a **do { } while()** loop or a **while() { }** loop
  - Returns **false** if there are no more rows
- typically we fetch a single row (mysql_fetch_assoc) followed by a **do { } while()** loop
  - to allow a "problem detection" code to run only once
  - or generate "single steps" for displaying a successful result (eg: table head)

Example

```
$hostname = "localhost";
$database = "world";
$username = "web";
$password = “ceva";
$conex= mysql_connect($hostname, $username, $password);
mysql_select_db($database, $ conex);
```

```
$query = "SELECT `Code`,`Name`,`Population` FROM `country` AS c ";
$result = mysql_query($ query, $ conex) or die(mysql_error());
$row_result = mysql_fetch_assoc($ result );
$totalRows_ result = mysql_num_rows($ result );
```

Accessing a MySQL server from PHP



!! IMPORTANT

PHP > 5.5

PHP 5.5, 7, 8

- Starting with PHP 5.5.0 the mysql extension was **deprecated**
  - any function from this extension generates an **E_DEPRECATED** error/warning
  - Starting with PHP 7.0.0 the mysql extension was **removed**
- Instead we must use:
  - mysqli extension (MySQL Improved)
  - PDO extension (PHP Data Objects)

mysqli extension

- Other than enhanced security offers access to newer facilities of the DB server:
  - Prepared Statements (speed, security)
    - server side
    - client side
  - server Stored Procedures (speed, security)
  - Multiple Statements
  - Transactions (integrity)

mysqli extension

- Supports two interfaces
  - procedural interfaces (similar to mysql)
  - OOP (similar to PDO)
- Procedural interface (almost) identical to the original mysql extension
  - easy transition
  - small differences (parameter)

mysqli – Procedural

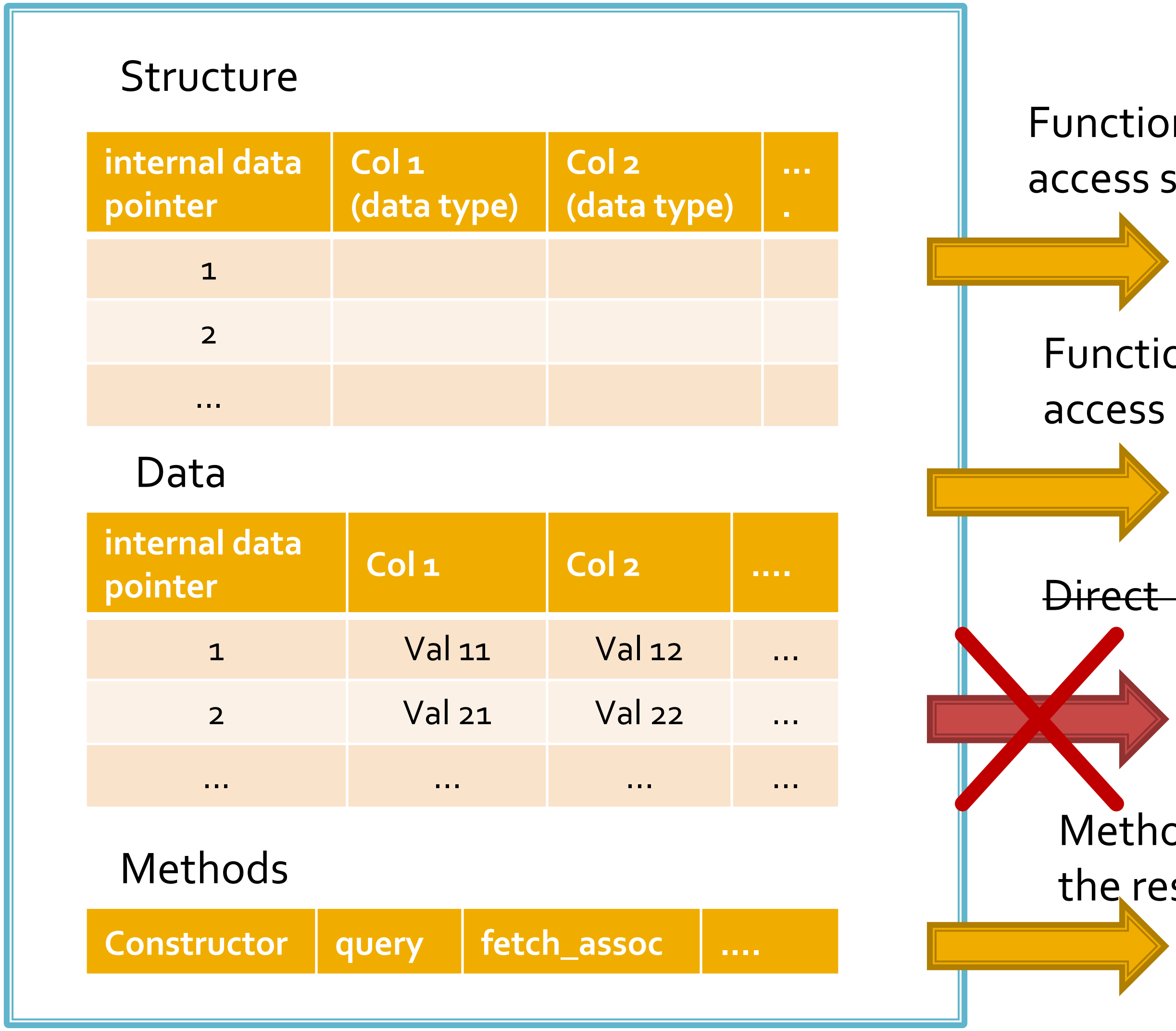
```
<?php
$mysqli = mysqli_connect("example.com", "user", "password", "database");
$res = mysqli_query($mysqli, "SELECT 'Please do not use the mysql extension ' AS _msg FROM DUAL");
$row = mysqli_fetch_assoc($res);
echo $row['_msg'];

$mysql = mysql_connect("example.com", "user", "password");
mysql_select_db("test");
$res = mysql_query("SELECT ' for new developments.' AS _msg FROM DUAL", $mysql);
$row = mysql_fetch_assoc($res);
echo $row['_msg'];
?>
```

- all mysql functions have a mysqli equivalent
- most functions have the same parameters in the same order
- there are functions with small differences (Ex: **mysqli_connect**, **mysqli_query**)

mysqli – OOP

```php
<?php
$var = new mysqli("example.com", "user", "password", "database");
$res = $var->query ($mysqli, "SELECT 'Please do not use the mysql extension ' AS _msg FROM DUAL");
$row = $res->fetch_assoc();
echo $row['_msg'];

$mysql = mysql_connect("example.com", "user", "password");
mysql_select_db("test");
$res = mysql_query("SELECT ' for new developments.' AS _msg FROM DUAL", $mysql);
$row = mysql_fetch_assoc($res);
echo $row['_msg'];
?>
```

MySQL resources – mysqli



Structure

internal data pointer	Col 1 (data type)	Col 2 (data type)	....
1			
2			
...			

Data

internal data pointer	Col 1	Col 2	....
1	Val 11	Val 12	...
2	Val 21	Val 22	...
...	...	...	...

Methods

Constructor	query	fetch_assoc	....

Functions to access structure

Functions to access data

~~Direct access~~

Methods attached to the resource

Conversion to mysqli (mandatory)

- example

```
$hostname = "localhost";
$database = "dbwpi";
$username = "web";
$password = “test";
$conex= mysql_connect($hostname, $username, $password);
mysql_select_db($database, $conex);

$query = "SELECT p.*, c.`nume` AS `nume_categ` FROM `produse` AS p
                LEFT JOIN `categorii` AS c ON (c.`id_categ` = p.`id_categ`)";
$result = mysql_query($query, $conex) or die(mysql_error());
$row_result = mysql_fetch_assoc($result);
$totalRows_result = mysql_num_rows($result);

do {
                $produse[$row_result['nume_categ']][$row_result['nume']]=array ("descr" => $row_result['detalii'], "pret"
=> $row_result['pret'], "cant" => $row_result['cant']);
                }
while ($row_result = mysql_fetch_assoc($result));
```

mysqli (Procedural)

```
//$conex= mysql_connect($hostname, $username, $password);
//mysql_select_db($database, $conex);
$conex = mysqli_connect($hostname, $username, $password, $database);

$query = "SELECT p.*, c.`nume` AS `nume_categ` FROM `produse` AS p
                LEFT JOIN `categorii` AS c ON (c.`id_categ` = p.`id_categ`)";
//$result = mysql_query($query, $conex) or die(mysql_error());
$result = mysqli_query($conex, $query);

//$row_result = mysql_fetch_assoc($result);
$row_result = mysqli_fetch_assoc($result);

//$totalRows_result = mysql_num_rows($result);
$totalRows_result = mysqli_num_rows($result);

do {
                $produse[$row_result['nume_categ']][$row_result['nume']]=array ("descr" => $row_result['detalii'], "pret"
=> $row_result['pret'], "cant" => $row_result['cant']);
                }
//while ($row_result = mysql_fetch_assoc($result));
while ($row_result = mysqli_fetch_assoc($result));
```

mysqli (OOP)

```
//$conex= mysql_connect($hostname, $username, $password);
//mysql_select_db($database, $conex);
//$conex = mysqli_connect($hostname, $username, $password, $database);
$conex = new mysqli($hostname, $username, $password, $database);

$query = "SELECT p.*, c.`nume` AS `nume_categ` FROM `produse` AS p
                LEFT JOIN `categorii` AS c ON (c.`id_categ` = p.`id_categ`)";
//$result = mysql_query($query, $conex) or die(mysql_error());
//$result = mysqli_query($conex, $query);
$result = $conex->query( $query );

//$row_result = mysql_fetch_assoc($result);
//$row_result = mysqli_fetch_assoc($result);
$row_result = $result->fetch_assoc();

//$totalRows_result = mysql_num_rows($result);
//$totalRows_result = mysqli_num_rows($result);
$totalRows_result = $result->num_rows;

do {
                $produse[$row_result['nume_categ']][$row_result['nume']]=array ("descr" => $row_result['detalii'], "pret"
=> $row_result['pret'], "cant" => $row_result['cant']);
                }
//while ($row_result = mysql_fetch_assoc($result));
while ($row_result = $result->fetch_assoc());
```

Textbooks

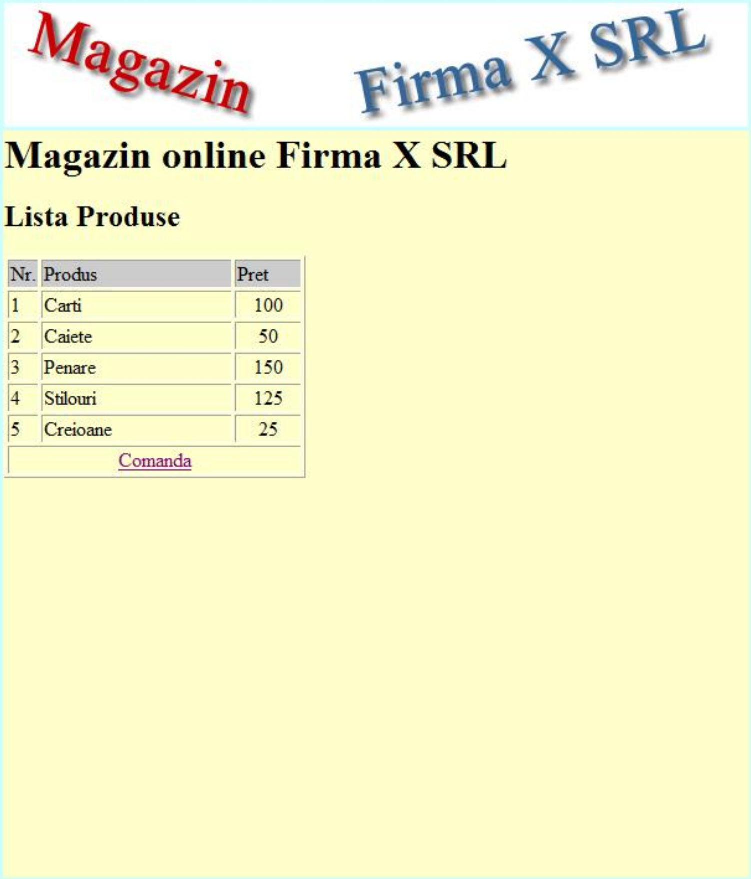
- https://www.php.net/
- https://rf-opto.etti.tuiasi.ro/master_it.php
    - **3 X PHP and MySQL Bible !!**

Template

Template

- simultaneous control of the esthetic and functional design for all pages in the site
- separation the application from the esthetic design

Example



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		

```
<html>
<head>
<title>Magazin online Firma X
SRL</title>
</head>
<body bgcolor="#CCFFFF">
<table width="600" border="0"
align="center">
<tr><td><img
src="images/antet.gif"
width="600" height="100"
/></td></tr>
<tr><td height="600" valign="top"
bgcolor="#FFFFCC">
Continut
</td></tr>
</table>
</body>
</html>
```

Control statements

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

- to insert the content of one PHP file (used as parameter) into another PHP file (that uses the include/require statement) before the server executes it
- **require** stops the execution of the current script if the parameter file is **not** found
- …**_once()** checks if the respective file has been included before and does not include it again

Example 2

- repeated sections can be moved to a separate file and inserted with require()
- first step: common areas are identified

```
<html>
<head>
<title>Magazin online Firma X SRL</title>
</head>
<body bgcolor="#CCFFFF">
<table width="600" border="0" align="center">
<tr><td><img src="images/antet.gif"
width="600" height="100" /></td></tr>
<tr><td height="600" valign="top"
bgcolor="#FFFFCC">
Continut
</td></tr>
</table>
</body>
</html>
```

Example 3

```
<html>
<head>
<title>Magazin online Firma X
SRL</title>
</head>
<body bgcolor="#CCFFFF">
<table width="600" border="0"
align="center">
<tr><td><img src="images/antet.gif"
width="600" height="100" /></td></tr>
<tr><td height="600" valign="top"
bgcolor="#FFFFCC">
Continut
</td></tr>
</table>
</body>
</html>
```

header.php

```
<html>
<head>
<title>Magazin online Firma X
SRL</title>
</head>
<body bgcolor="#CCFFFF"><?php
//orice cod comun PHP
?><table width="600" border="0"
align="center">
<tr><td><img src="images/antet.gif"
width="600" height="100" /></td></tr>
<tr><td height="600" valign="top"
bgcolor="#FFFFCC">
<h1>Magazin online Firma X SRL</h1>
```

footer.php

```
</td></tr>
</table>
</body>
</html>
```

Using a template

- header.php
  - any common structure (HTML) code
  - any common application code (PHP) – almost all pages in an application need:
    - data access
    - check access rights
    - constant definitions
    - define/load data **from** session ($_SESSION)
- footer.php
  - any common structure (HTML) code
  - any common application code (PHP) – usually less:
    - save data **into** the session ($_SESSION)

Template

- Any php file in my application:
  - <?php require('header.php');?>
  - <?php require('footer.php');?>
- and automatically that file has the same esthetic and functional design

*.php

```
<?php require('header.php');?>
<h2>Lista Produse</h2>
<table border="1">
...
</table>
<?php require('footer.php');?>
```

Advantages working with template

- speed of application development
- clear separation of the application from the form
- unitary form
  - “don’t make me think”
- simultaneous control of the esthetic and functional design for all pages in the site
- defining common data in a single file
  - define('BOOK_PRICE',100);

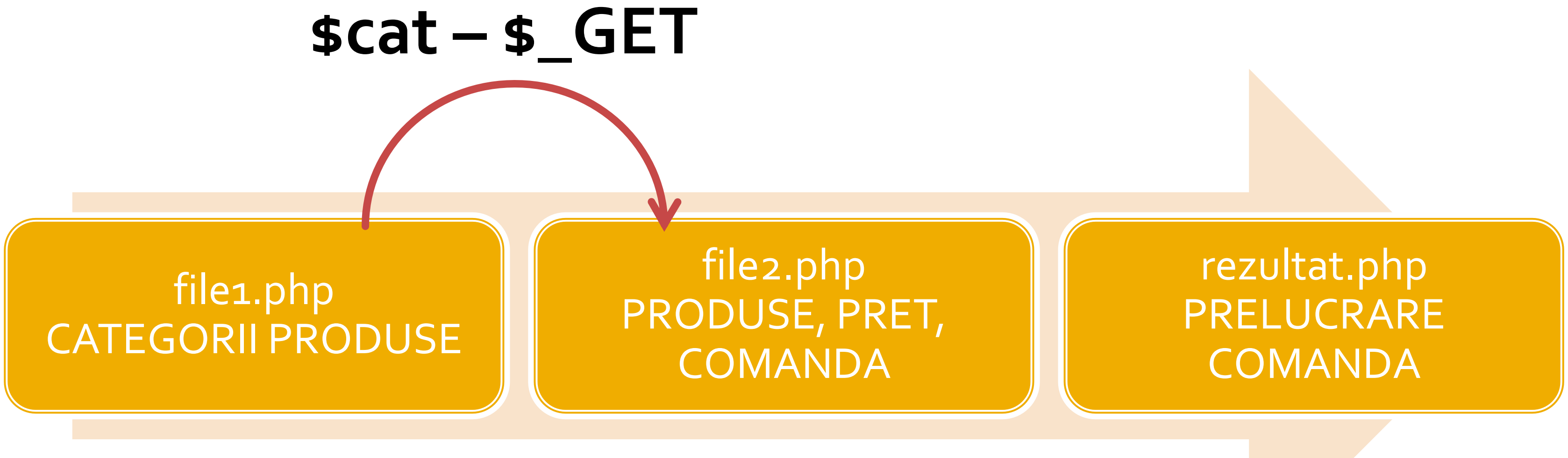
Active Links

Methods

- **post** : data is transmitted as a block (inside the body of the HTTP request)
- **get** : appends form-data into the URL : results.php?prob=81&an=2009
- **get** must be used only for "idempotent" data,
  - no collateral effects
  - no change in server status (databases, etc)
- we can emulate a form (**get**) by writing links appropriately

Active Links

- used to send specific **information** to the target file
- in file1.php
  - <a href="file2.php?categ=<?php echo $cat;?>"> <?php echo $cat;?> </a>
- in file2.php
  - $_GET['categ']="value $cat associated to that specific link"



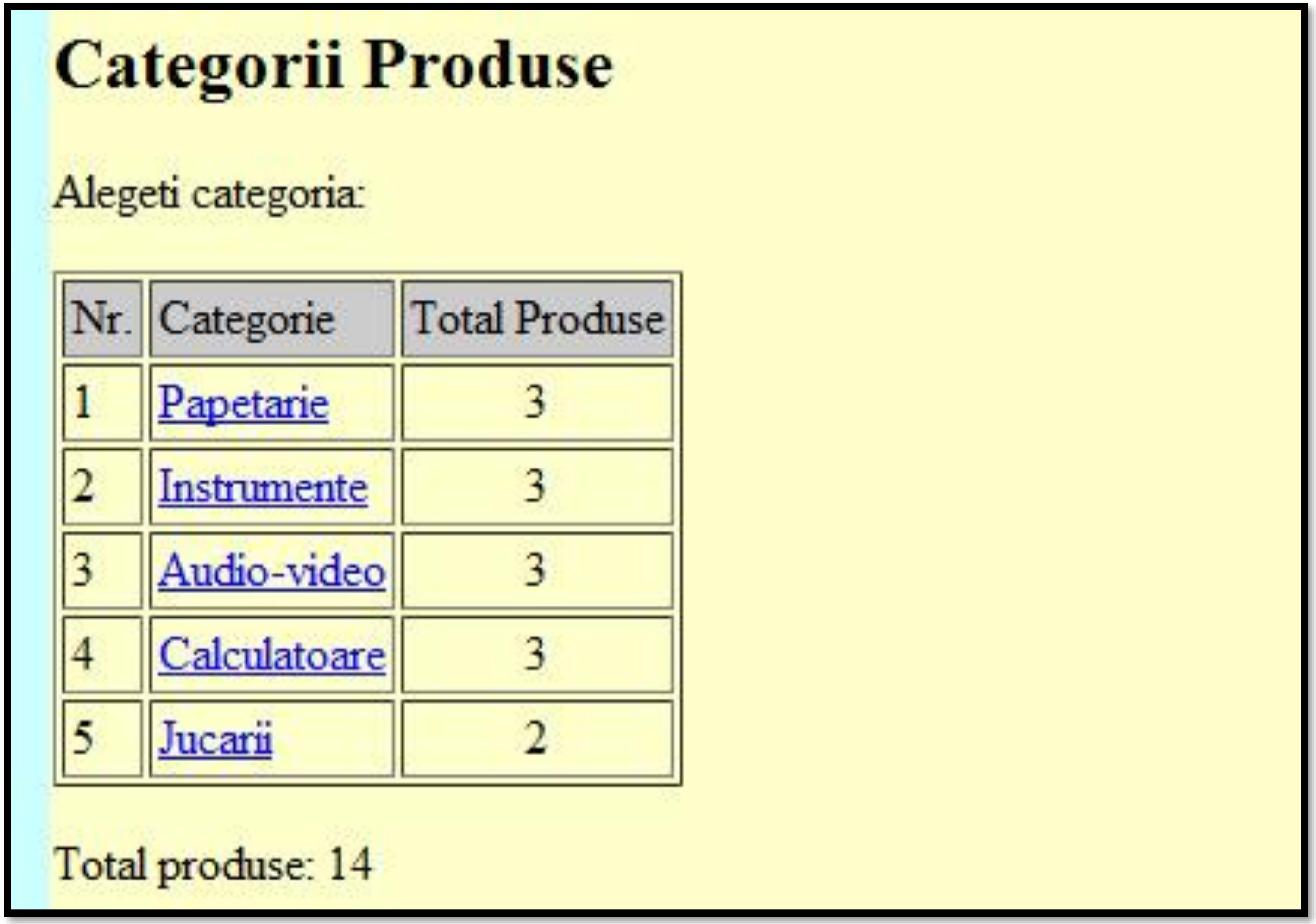
$cat – $_GET

file1.php
CATEGORII PRODUSE

file2.php
PRODUSE, PRET, COMANDA

rezultat.php
PRELUCRARE COMANDA

Active Links



Categorii Produse

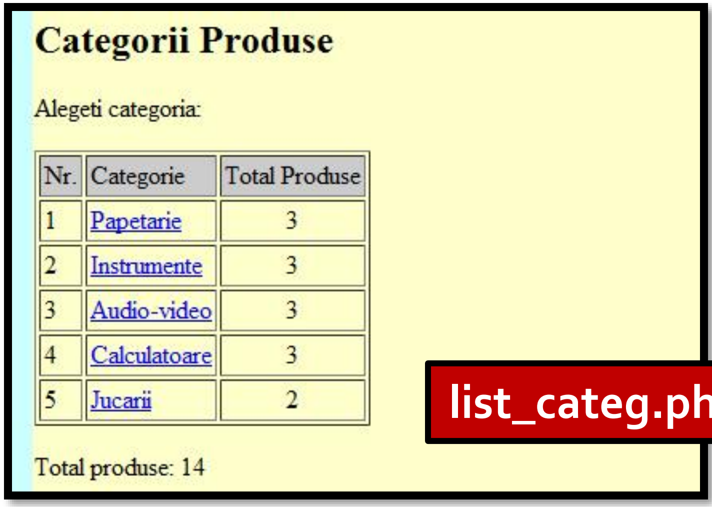
Alegeti categoria:

Nr.	Categorie	Total Produse
1	Papetarie	3
2	Instrumente	3
3	Audio-video	3
4	Calculatoare	3
5	Jucarii	2

Total produse: 14

Application flowchart

Rudimentary online shop app



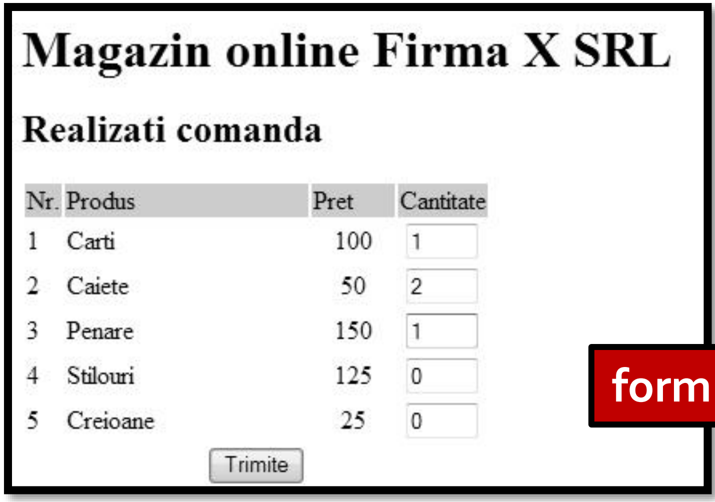
Categorii Produse

Alegeti categoria:

Nr.	Categorie	Total Produse
1	Papetarie	3
2	Instrumente	3
3	Audio-video	3
4	Calculatoare	3
5	Jucarii	2

Total produse: 14

list_categ.php

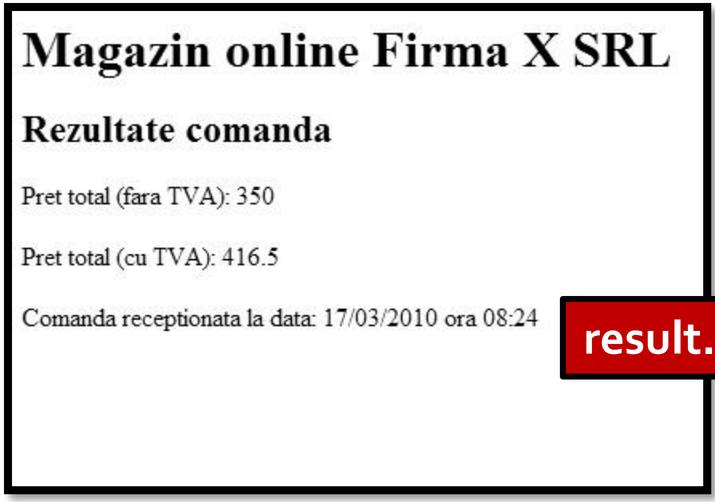


Magazin online Firma X SRL

Realizati comanda

Nr.	Produs	Pret	Cantitate
1	Carti	100	1
2	Caiete	50	2
3	Penare	150	1
4	Stilouri	125	0
5	Creioane	25	0

form.php



Magazin online Firma X SRL

Rezultate comanda

Pret total (fara TVA): 350

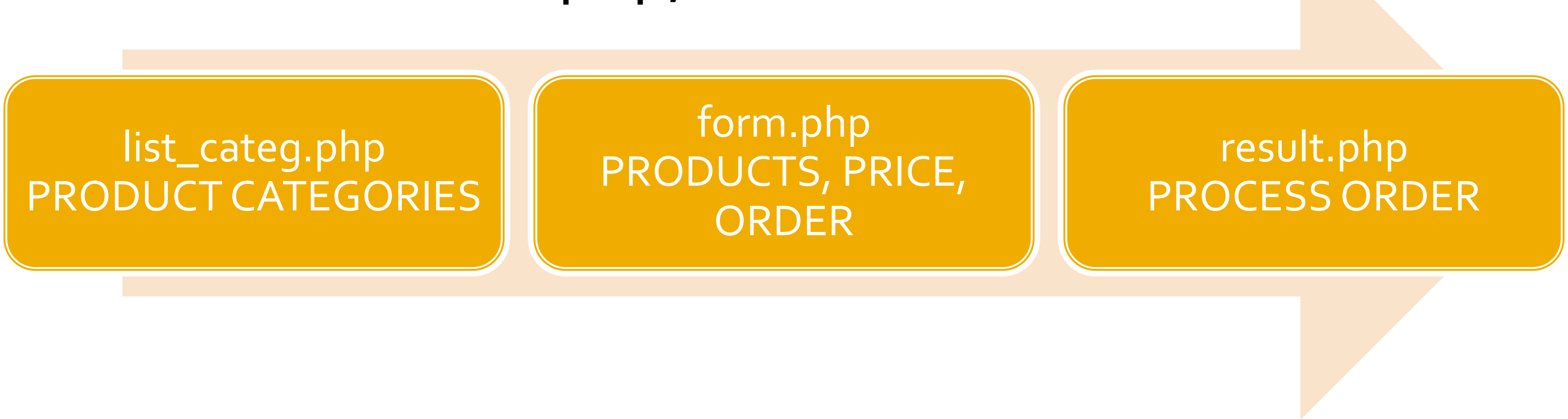
Pret total (cu TVA): 416.5

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

result.php

Application flowchart – Buyer

- As the application leaves a linear thread of execution, it is necessary to introduce a flowchart (tree) of the application
- Buyer
  - reading the required data (database access) is done in header.php, common for all files



list_categ.php
PRODUCT CATEGORIES

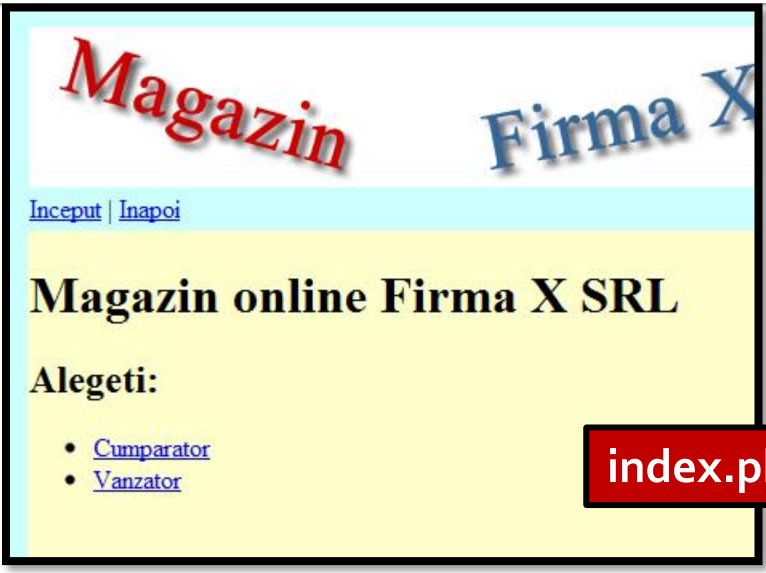
form.php
PRODUCTS, PRICE, ORDER

result.php
PROCESS ORDER

Application flowchart – Seller

- The appearance of the application for the seller
  - introduces a parallel thread of execution with the necessity of the initial choice: buyer/seller
  - brings the possibility of writing in the database
  - various writing operations
    - insert new product category
    - insert new product in an existing category
    - modify existing product
  - writing in the database involves 2 actions:
    - collection of raw data from the user
    - data processing

Seller thread for online shop app



Magazin Firma X

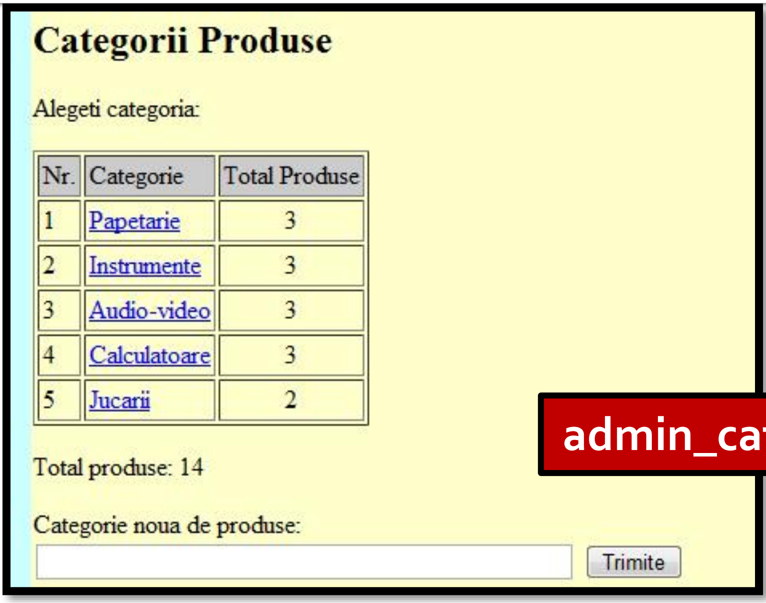
Inceput | Inapoi

Magazin online Firma X SRL

Alegeti:

- Cumparator
- Vanzator

index.php



Categorii Produse

Alegeti categoria:

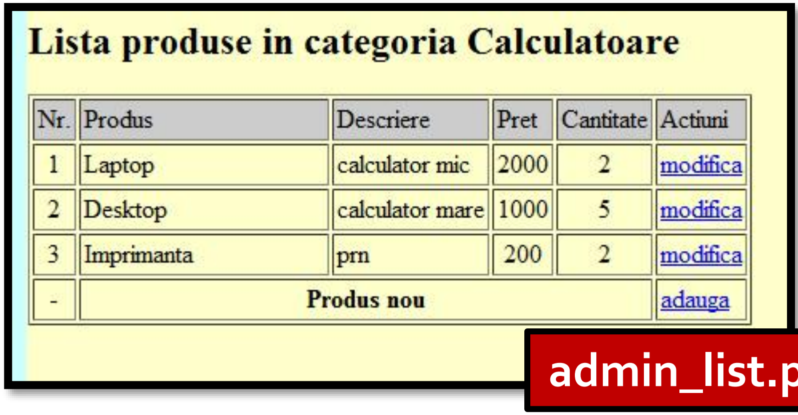
Nr.	Categorie	Total Produse
1	Papetarie	3
2	Instrumente	3
3	Audio-video	3
4	Calculatoare	3
5	Jucarii	2

Total produse: 14

Categorie noua de produse:

Trimite

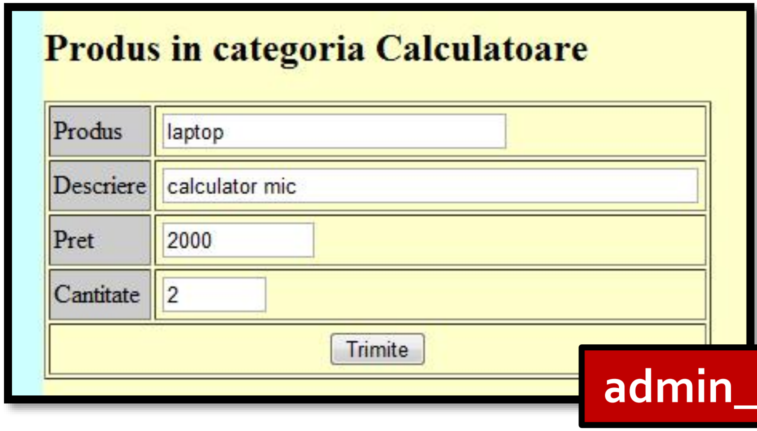
admin_categ.php



Lista produse in categoria Calculatoare

Nr.	Produs	Descriere	Pret	Cantitate	Actiuni
1	Laptop	calculator mic	2000	2	modifica
2	Desktop	calculator mare	1000	5	modifica
3	Imprimanta	prn	200	2	modifica
-	**Produs nou**				adauga

admin_list.php

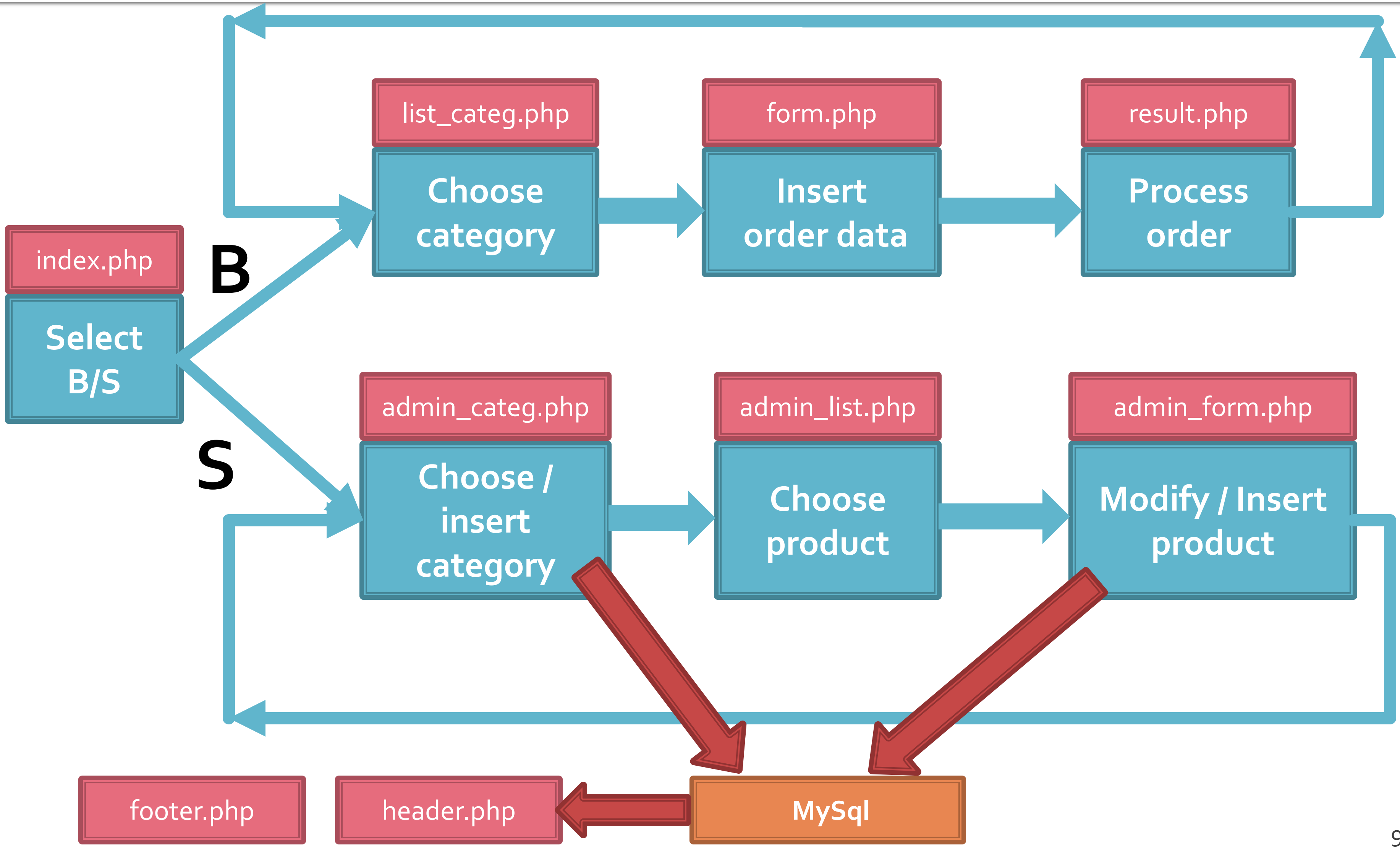


Produs in categoria Calculatoare

Produs	laptop
Descriere	calculator mic
Pret	2000
Cantitate	2
	Trimite

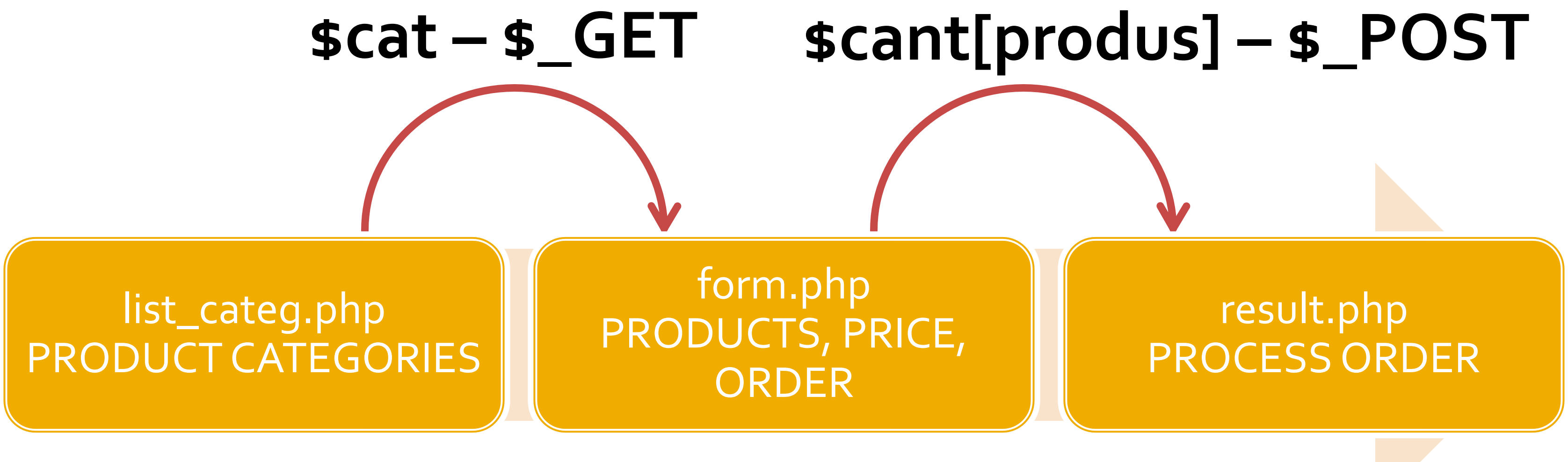
admin_form.php

Application flowchart



Application flowchart

- The application flowchart must also include information related to :
  - **what** data is transmitted between the different pages
  - **how** data is transmitted between pages



$cat – $_GET

$cant[produs] – $_POST

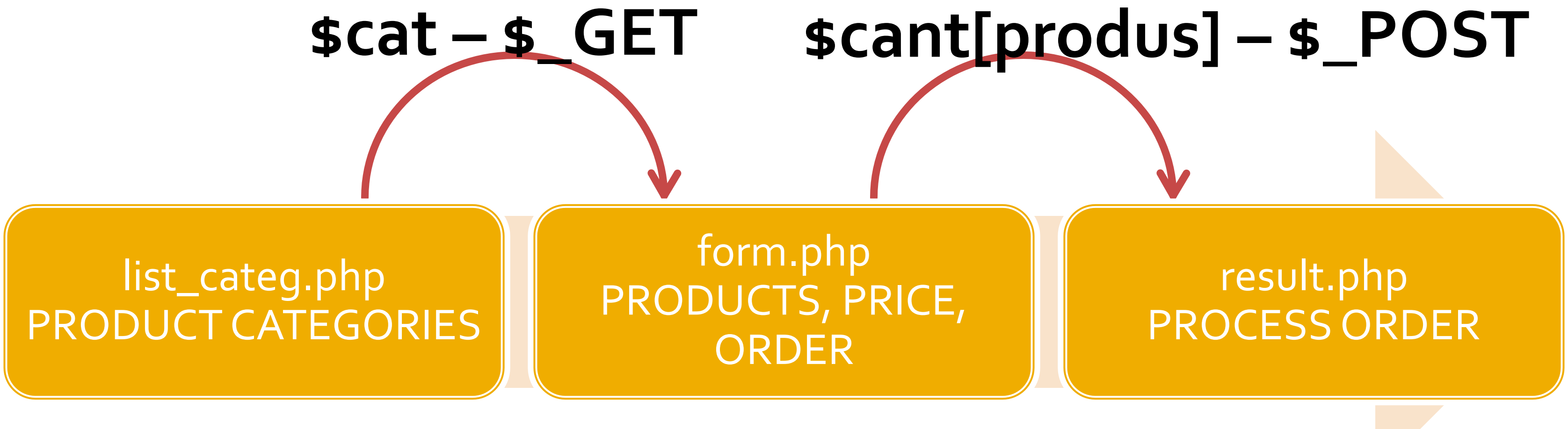
list_categ.php
PRODUCT CATEGORIES

form.php
PRODUCTS, PRICE, ORDER

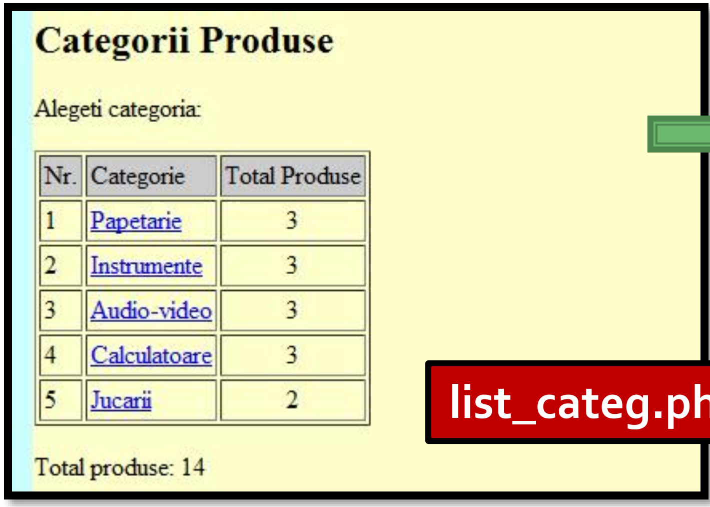
result.php
PROCESS ORDER

Application flowchart

- Application flowchart – Example
  - the list of categories will transmit a single variable to the next file so we can use "active links", get method, **$_GET** in next file
  - the order form transmits multiple data included in a form, so the transmission is done with post method, **$_POST** in next file
- The choice of $_GET/$_POST has implications both in:
  - the page that transmits the data
  - as well as on the page that receives them



Flowchart (Buyer)



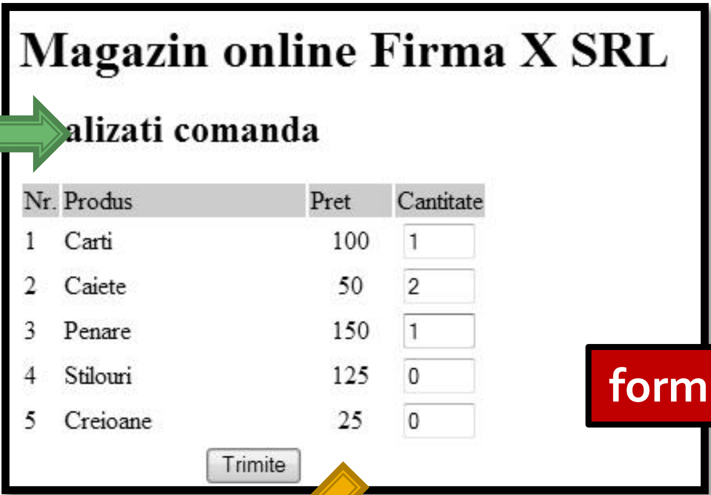
Categorii Produse

Alegeti categoria:

Nr.	Categorie	Total Produse
1	Papetarie	3
2	Instrumente	3
3	Audio-video	3
4	Calculatoare	3
5	Jucarii	2

Total produse: 14

list_categ.php

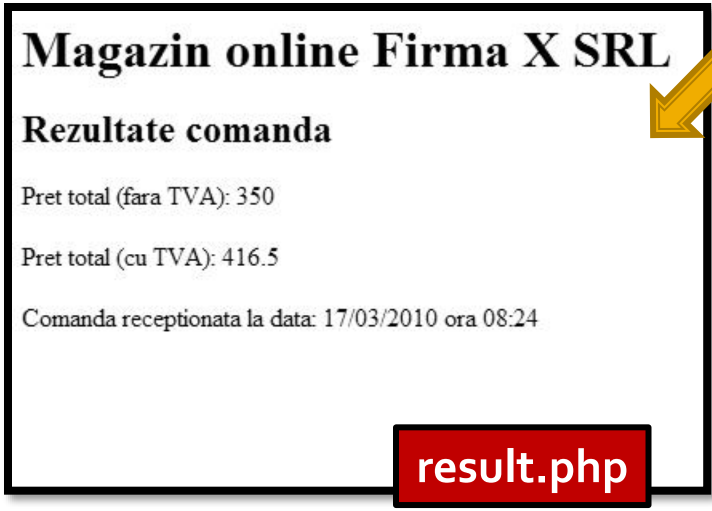


Magazin online Firma X SRL

alizati comanda

Nr.	Produs	Pret	Cantitate
1	Carti	100	1
2	Caiete	50	2
3	Penare	150	1
4	Stilouri	125	0
5	Creioane	25	0

form.php



Magazin online Firma X SRL

Rezultate comanda

Pret total (fara TVA): 350

Pret total (cu TVA): 416.5

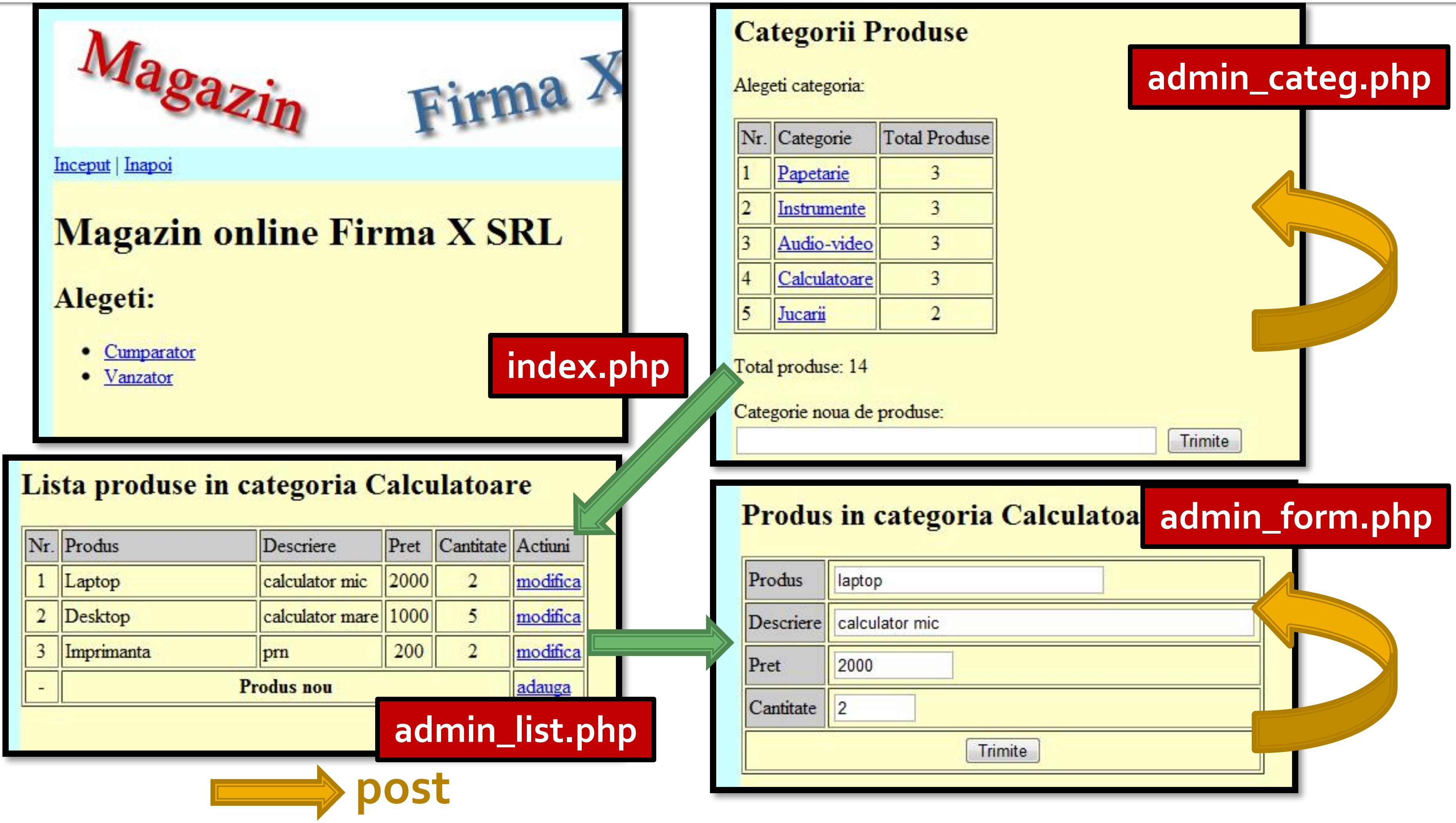
Comanda receptionata la data: 17/03/2010 ora 08:24

result.php

post

get

Flowchart (Seller)



index.php

Magazin Firma X

Inceput | Inapoi

Magazin online Firma X SRL

Alegeti:

- Cumparator
- Vanzator

admin_categ.php

Categorii Produse

Alegeti categoria:

Nr.	Categorie	Total Produse
1	Papetarie	3
2	Instrumente	3
3	Audio-video	3
4	Calculatoare	3
5	Jucarii	2

Total produse: 14

Categorie noua de produse:

Trimite

admin_list.php

Lista produse in categoria Calculatoare

Nr.	Produs	Descriere	Pret	Cantitate	Actiuni
1	Laptop	calculator mic	2000	2	modifica
2	Desktop	calculator mare	1000	5	modifica
3	Imprimanta	prn	200	2	modifica
-	Produs nou				adauga

admin_form.php

Produs in categoria Calculatoa

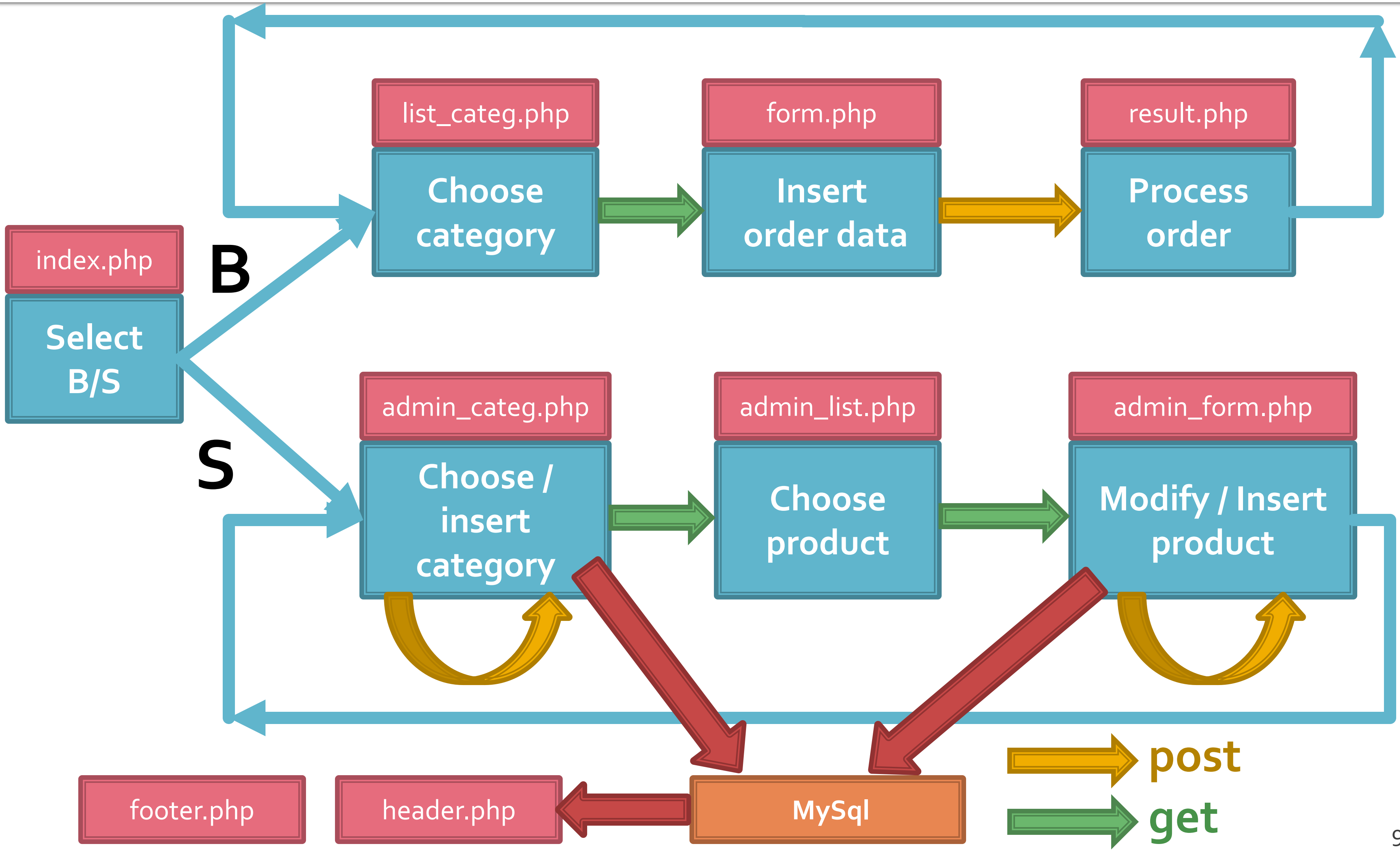
Produs	laptop
Descriere	calculator mic
Pret	2000
Cantitate	2

Trimite

post

get

Complete application flowchart



index.php

Select B/S

B

S

list_categ.php

Choose category

form.php

Insert order data

result.php

Process order

admin_categ.php

Choose / insert category

admin_list.php

Choose product

admin_form.php

Modify / Insert product

footer.php

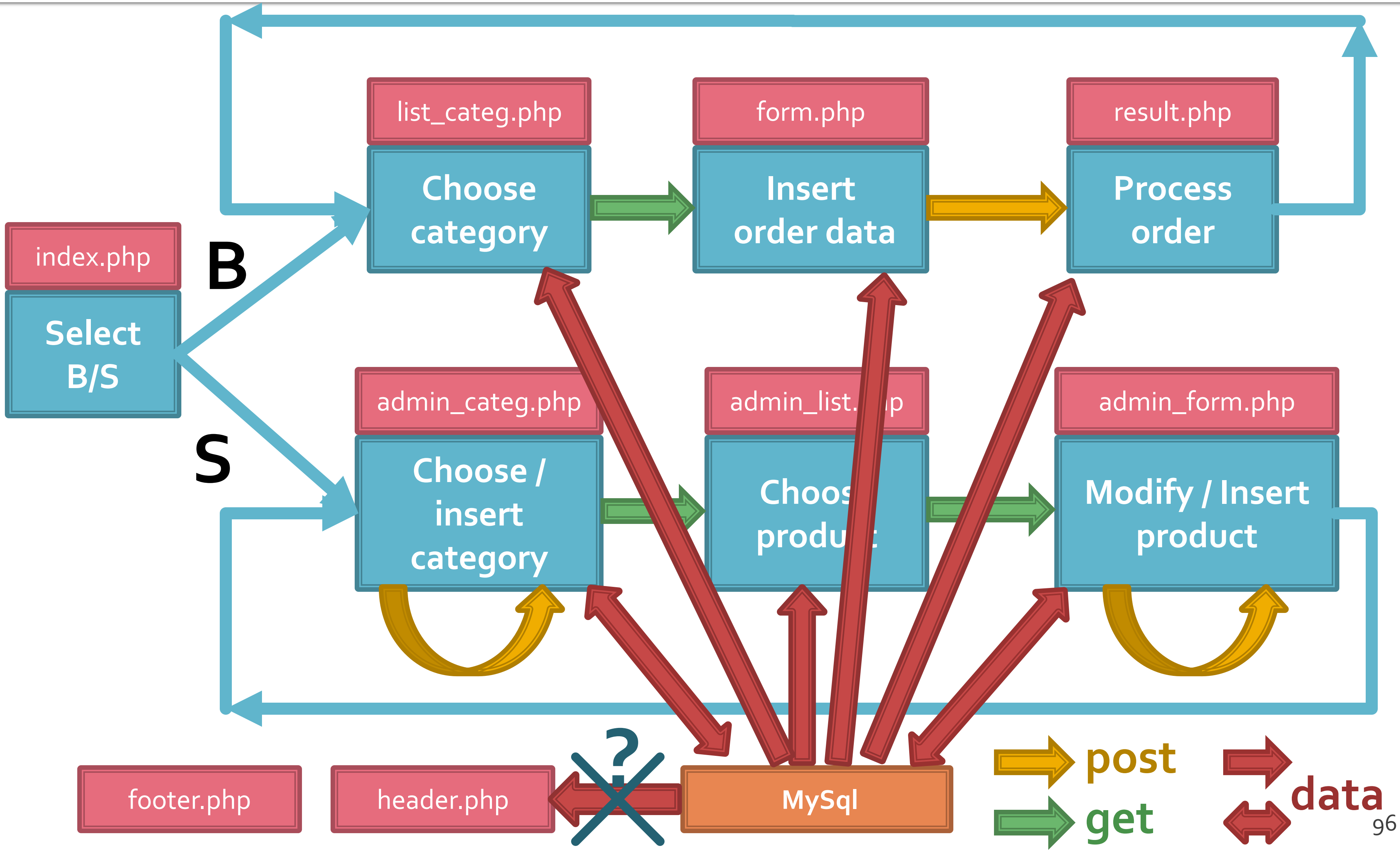
header.php

MySql

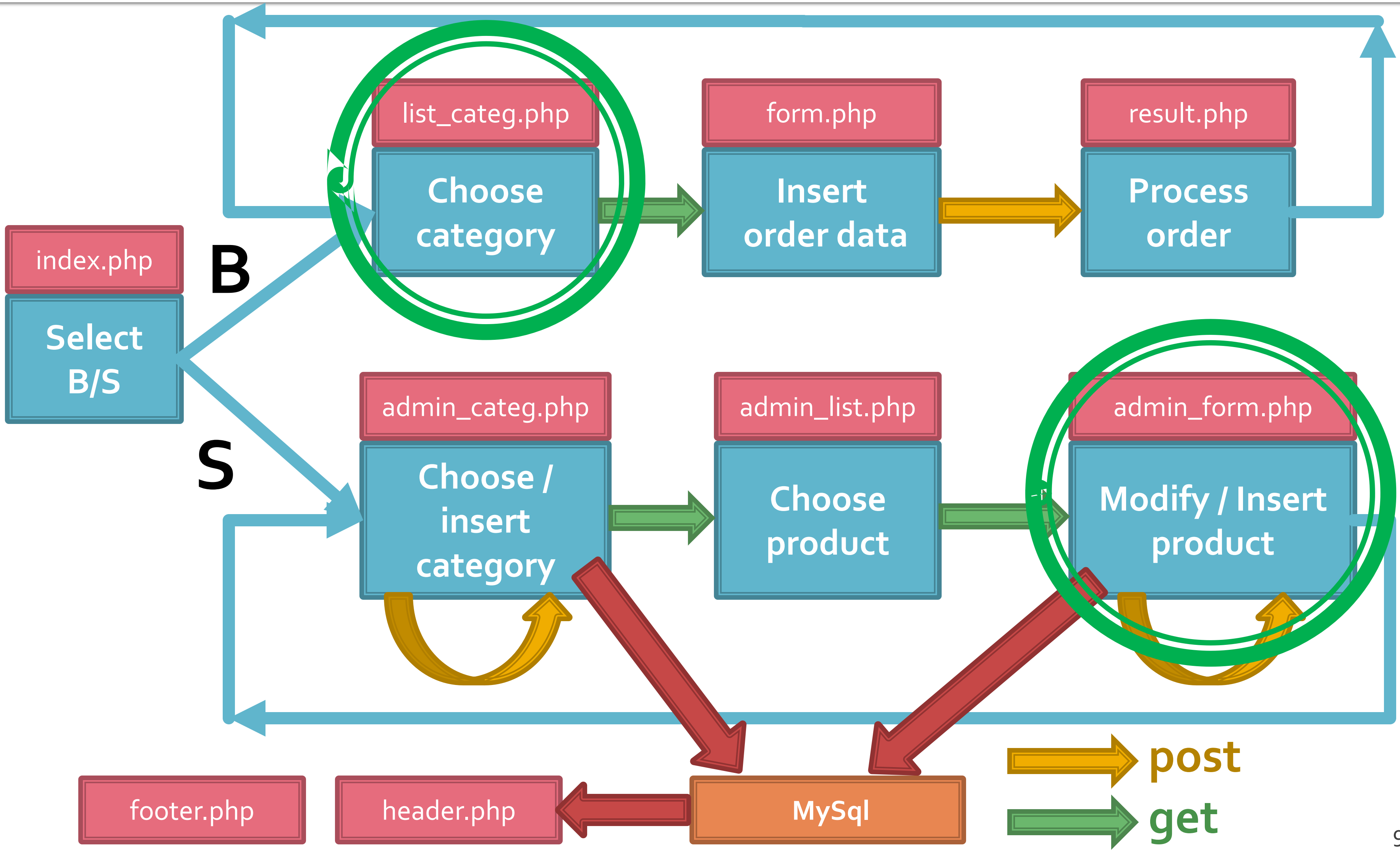
post

get

Optimal application flowchart



Individual assignment



index.php

Select B/S

B

S

list_categ.php

Choose category

form.php

Insert order data

result.php

Process order

admin_categ.php

Choose / insert category

admin_list.php

Choose product

admin_form.php

Modify / Insert product

footer.php

header.php

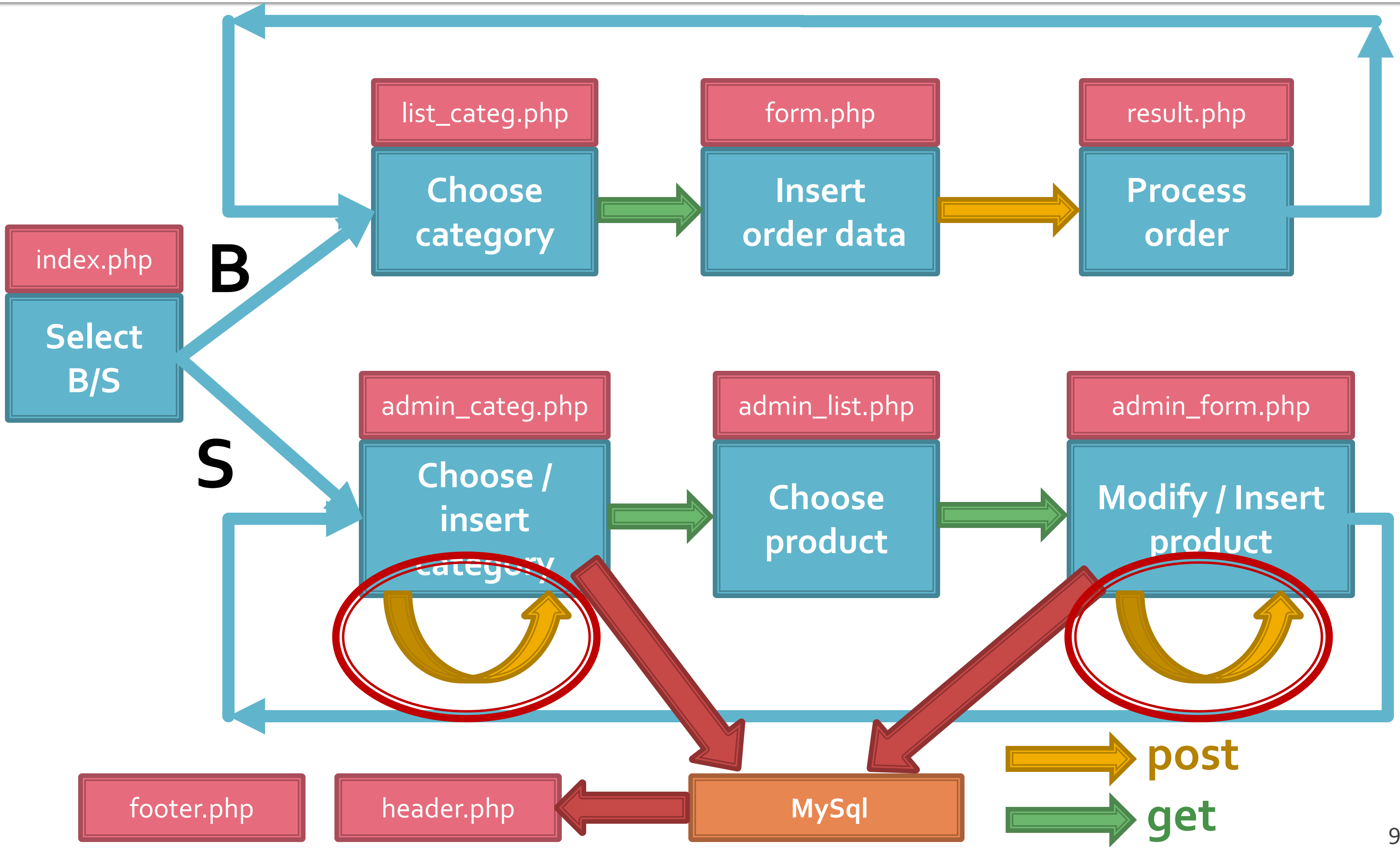
MySql

post

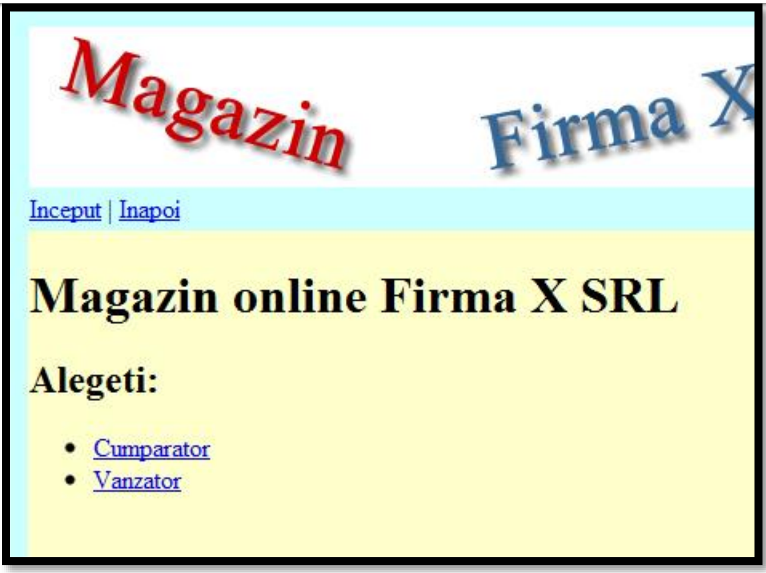
get

Single file for data collection AND processing

Complete application flowchart



Flowchart (Seller)

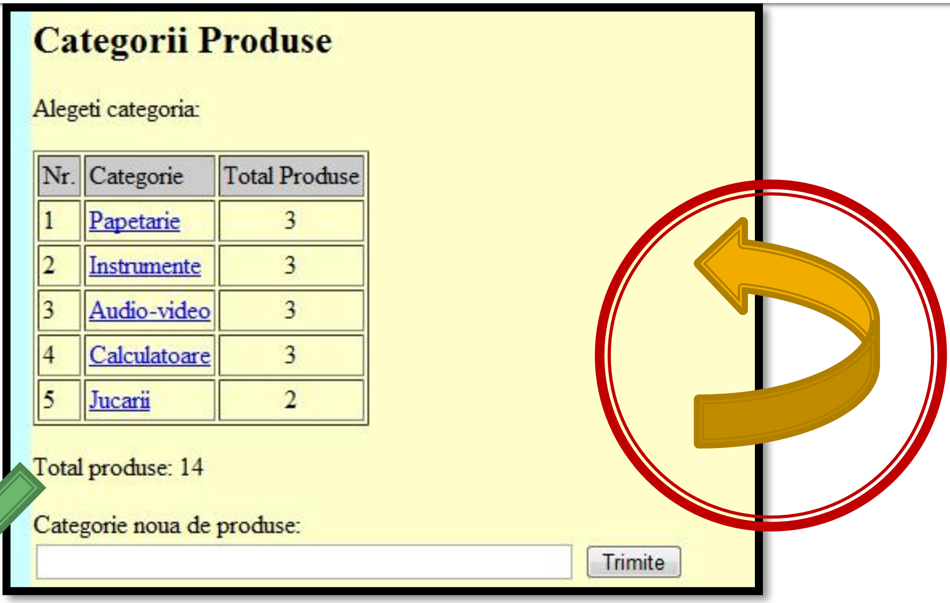


Inceput | Inapoi

Magazin online Firma X SRL

Alegeti:

- Cumparator
- Vanzator



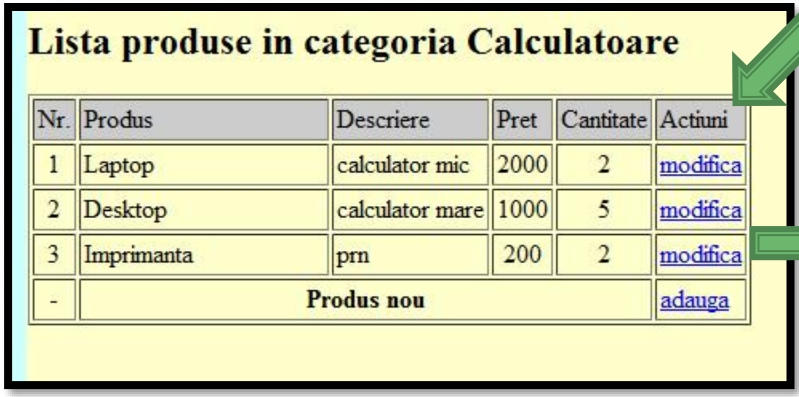
Categorii Produse

Alegeti categoria:

Nr.	Categorie	Total Produse
1	Papetarie	3
2	Instrumente	3
3	Audio-video	3
4	Calculatoare	3
5	Jucarii	2

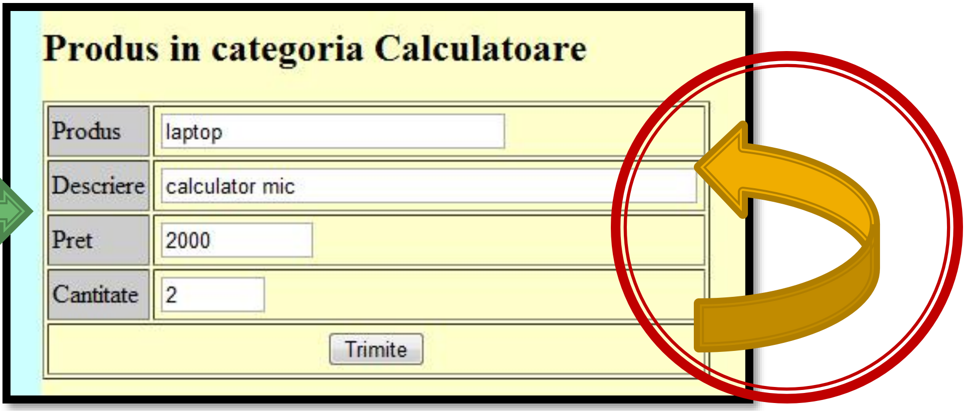
Total produse: 14

Categorie noua de produse:



Lista produse in categoria Calculatoare

Nr.	Produs	Descriere	Pret	Cantitate	Actiuni
1	Laptop	calculator mic	2000	2	modifica
2	Desktop	calculator mare	1000	5	modifica
3	Imprimanta	prn	200	2	modifica
-	**Produs nou**				adauga



Produs in categoria Calculatoare

Produs	laptop
Descriere	calculator mic
Pret	2000
Cantitate	2

post

get

Single file for data collection AND processing

- This option is often preferred
- It allows the unitary preservation of all operations for the performance of an action
  - easier access
  - ease of programming
  - avoiding errors: File does not exist: /Server/…
- The same file is initially used to collect data and then, if their presence is detected, for their processing

Single file for data collection AND processing

- The "action" file for <form> will be the current file
- it is recommended to use the global variable $_SERVER['SCRIPT_NAME']
  - flexibility when renaming files
- alternatively $_SERVER['PHP_SELF'] is not recommended
  - security issues
- The data collection section is displayed only in the absence of data

```
<form action="<?php echo $_SERVER['SCRIPT_NAME '];?>" method="post">
<p><input name=“date_ok" type="submit" value="Trimite" /></p>
</form>
```

Single file for data collection AND processing

- The detection of the existence of the data is done by checking the existence (**isset**($variable)) of the user inserted values
  - for extra protection, their content can also be checked

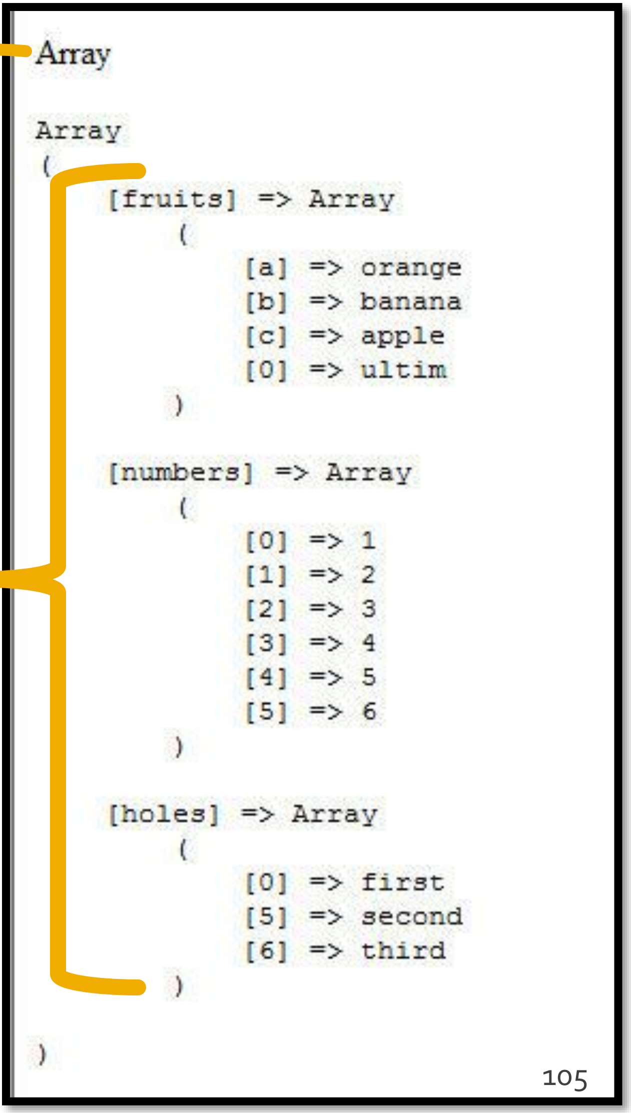
```
if (isset($_POST[" date_ok "]))
        { //date trimise
        if ($_POST[" date_ok "]=="Trimite" )
                { // data sent by the current file
                // data processing
                }
        }
else
        {
        // data collection
<form action="<?php echo $_SERVER['SCRIPT_NAME '];?>" method="post">
<p><input name="date_ok" type="submit" value="Trimite" /></p></form>
        }
```

PHP Debug

View array content (debug)

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

```
$matr= array (
"fruits"  =>
array("a" => "orange", "b" => "banana", "c" => "apple" ,
"ultim"),
"numbers" =>
array(1, 2, 3, 4, 5, 6),
"holes"   =>
array("first", 5 => "second", "third")
);
echo $matr;
echo "<pre>";
print_r ($matr);
echo "</pre>";
```



```
Array

Array
(
    [fruits] => Array
        (
            [a] => orange
            [b] => banana
            [c] => apple
            [0] => ultim
        )

    [numbers] => Array
        (
            [0] => 1
            [1] => 2
            [2] => 3
            [3] => 4
            [4] => 5
            [5] => 6
        )

    [holes] => Array
        (
            [0] => first
            [5] => second
            [6] => third
        )

)
```

Verify/debug PHP code

- It is recommended to use the array visualization option
  - In the file that receives the data
  - temporarily until the final version of the code

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

- the use of "verbose" code (manual) in the initial stages of writing PHP code can be extended to other types of data
  - the only (almost) debugging method in PHP
  - <p>temp <?php echo "a=";echo $a; ?> </p>

Debug

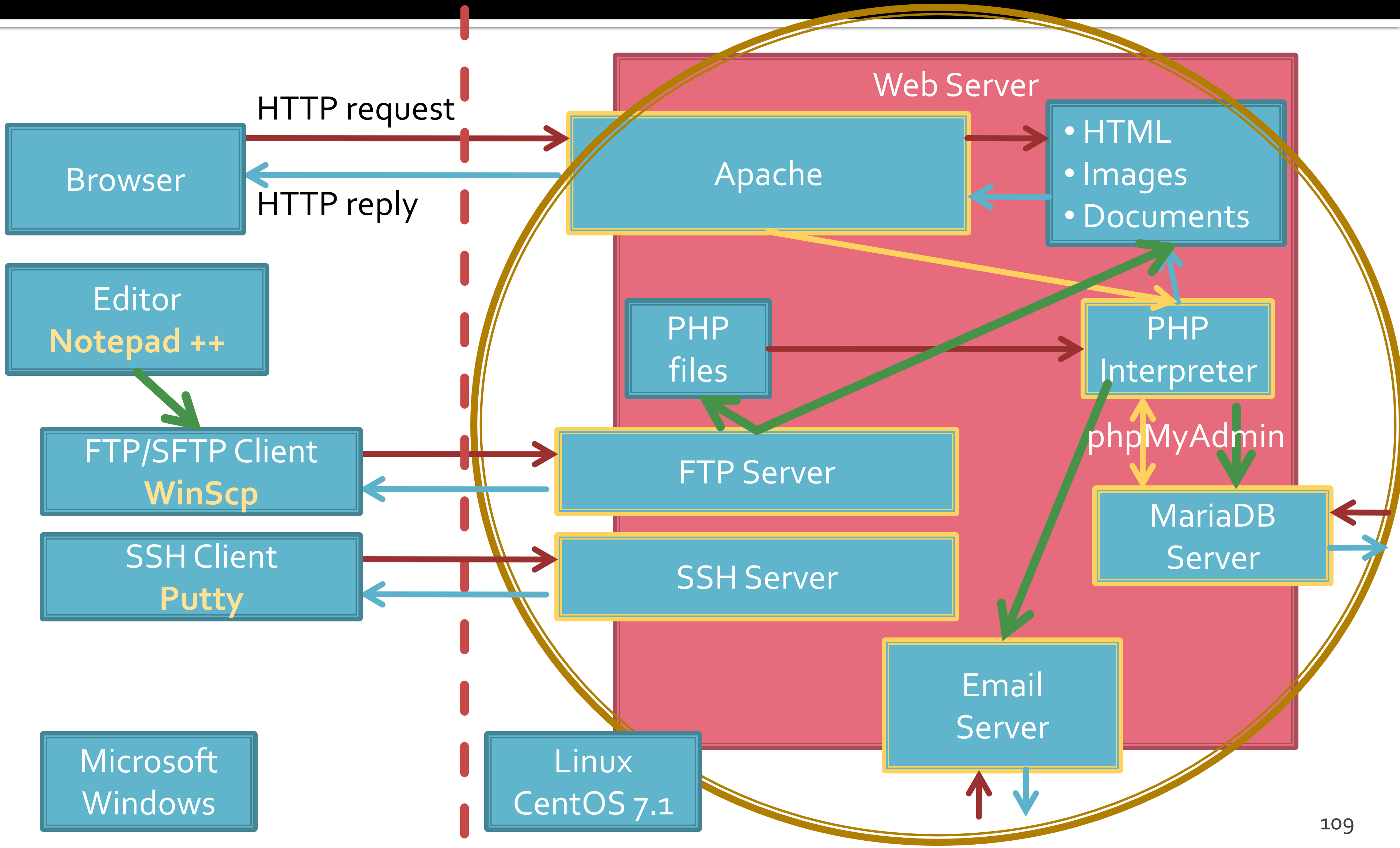
```
echo "<pre>";
print_r ($_POST);
echo "</pre>";
```

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

```
echo "<pre>".print_r ($_GET,true)."</pre>";
```

Reference server and applications

Using LAMP


Web Server
HTTP request
HTTP reply
Browser
Apache
• HTML
• Images
• Documents
Editor
Notepad ++
PHP
files
PHP
Interpreter
phpMyAdmin
FTP/SFTP Client
WinScp
FTP Server
MariaDB
Server
SSH Client
Putty
SSH Server
Email
Server
Microsoft
Windows
Linux
CentOS 7.1

Using LAMP

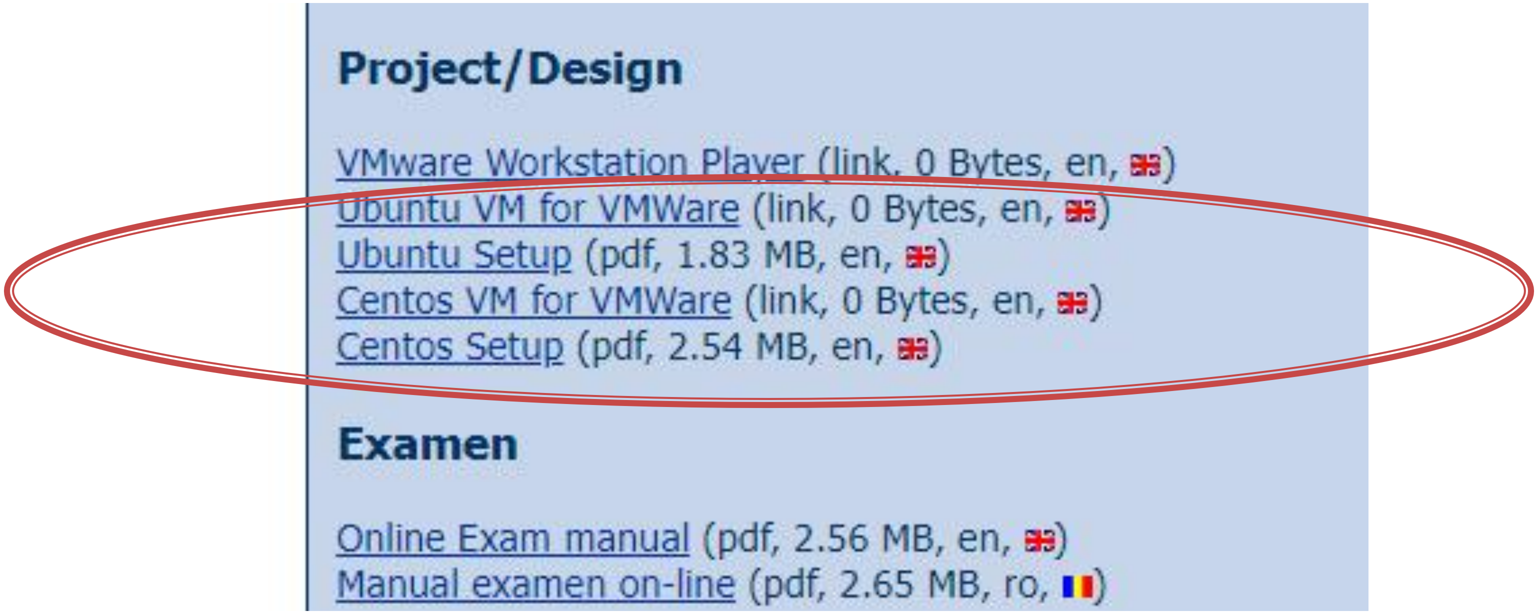
- differences from a Windows computer
  - system commands difficult
    - command prompt, SSH, Putty
  - files submitted by FTP
    - Copy/Paste unavailable
  - administration of MySql server:
    - using PhpMyAdmin (preloaded)
    - using other software on the host computer (eg. MySQL Workbench)

Using LAMP – Advantages

- Advantages
  - Available applications have newer versions (**2020 ~**)
    - CentOS/7.1, Apache/2.4.6, PHP/5.4.16, MariaDB/5.5.44, PhpMyAdmin/4.4.15
    - Ubuntu/20.04, Apache/ 2.4.41, PHP/ 7.4.3, MariaDB/ 10.3.31, PhpMyAdmin/4.9.5
    - Debian/12.5, Apache/ 2.4.57, PHP/ 8.2.7, MariaDB/ 10.11.6, PhpMyAdmin/5.2.1
  - Available applications very similar with real life hosting solutions
    - SSH
    - FTP
    - Email
      - for full functionality the network interface for the VM must be changed **Host-only** -> **Bridged**

Reference Server

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



Project/Design

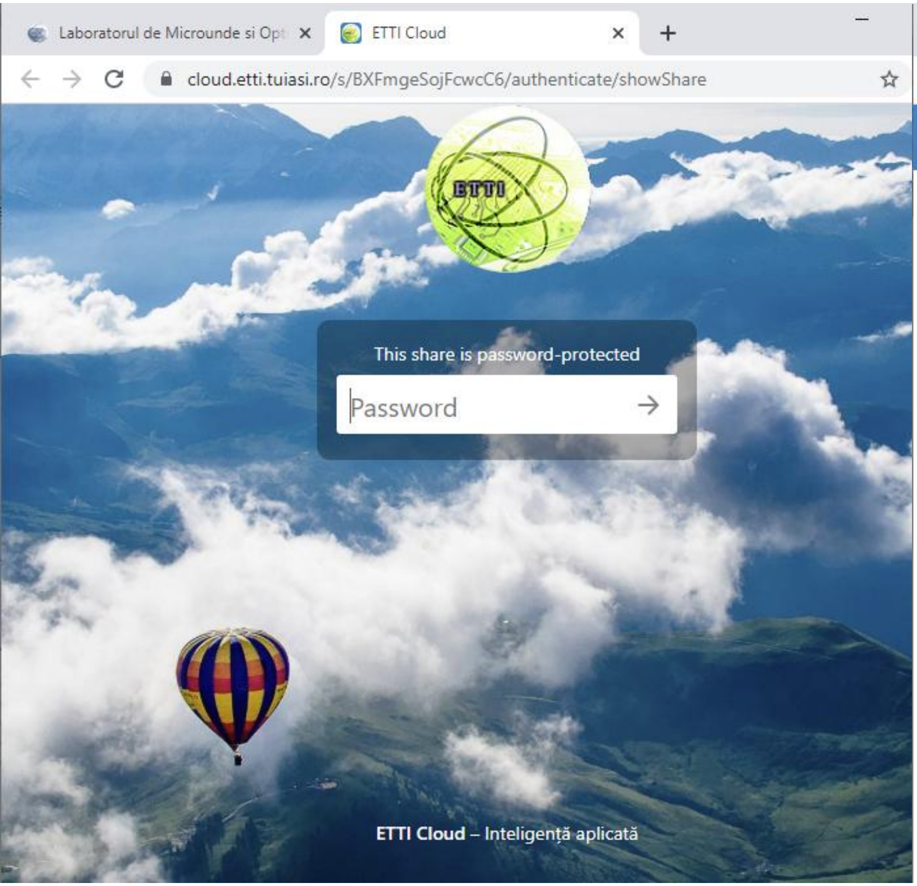
VMware Workstation Player (link, 0 Bytes, en, 🇬🇧)
Ubuntu VM for VMWare (link, 0 Bytes, en, 🇬🇧)
Ubuntu Setup (pdf, 1.83 MB, en, 🇬🇧)
Centos VM for VMWare (link, 0 Bytes, en, 🇬🇧)
Centos Setup (pdf, 2.54 MB, en, 🇬🇧)

Examen

Online Exam manual (pdf, 2.56 MB, en, 🇬🇧)
Manual examen on-line (pdf, 2.65 MB, ro, 🇷🇴)

Reference Server

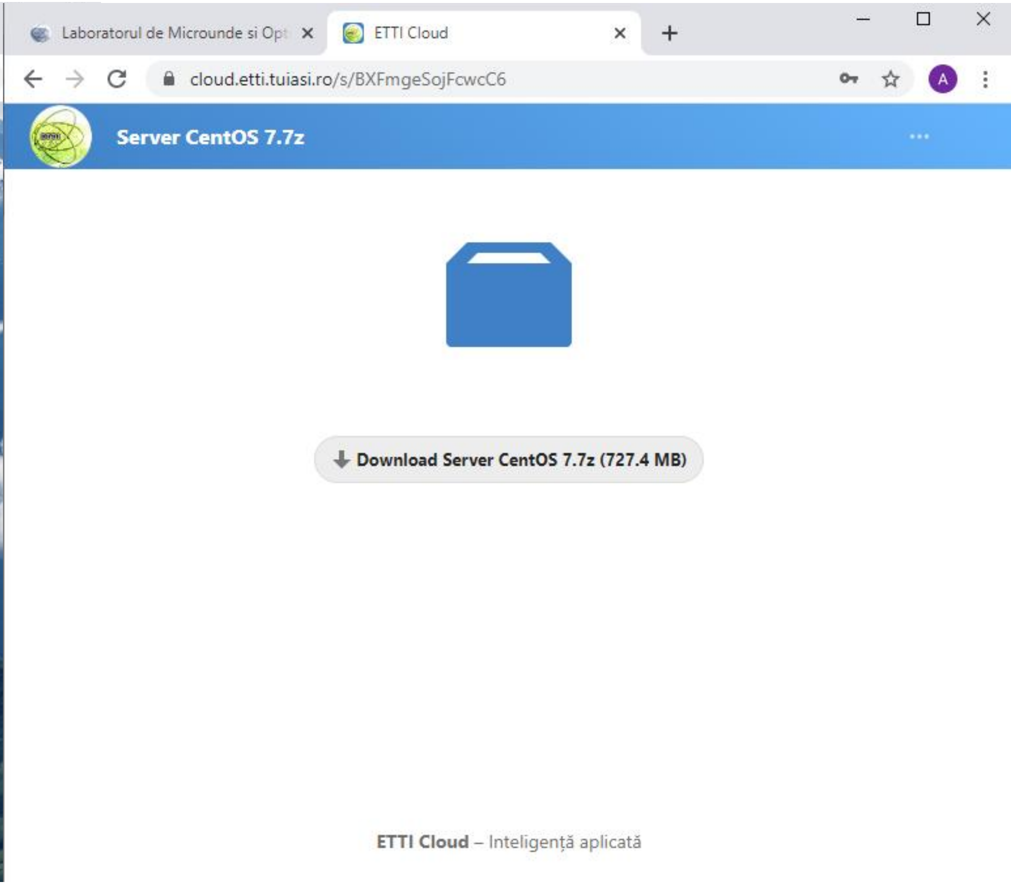
- Cloud ETTI: **RF-opto3#**



cloud.etti.tuiasi.ro/s/BXFmgeSojFcwcC6/authenticate/showShare

This share is password-protected

ETTI Cloud – Inteligență aplicată



Laboratorul de Microunde si Opt
ETTI Cloud
A

cloud.etti.tuiasi.ro/s/BXFmgeSojFcwcC6

Server CentOS 7.7z

Download Server CentOS 7.7z (727.4 MB)

ETTI Cloud – Inteligență aplicată

Server referinta

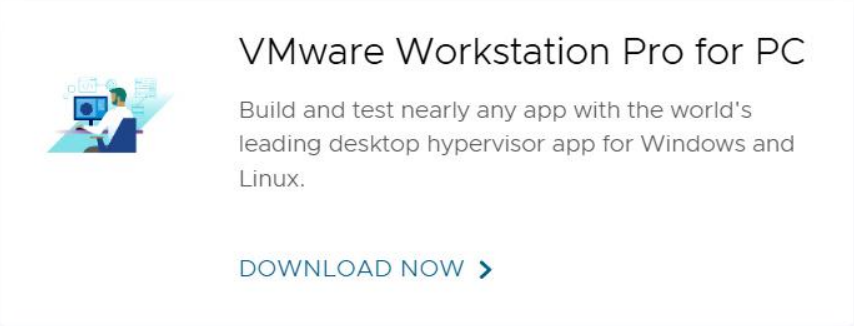
- Masina virtuala
- VMware Workstation Player
  - Gratuit (non-comercial)
  - https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html
- Inlocuit de VMware Workstation Pro

VMware Workstation Pro for PC

Build and test nearly any app with the world's leading desktop hypervisor app for Windows and Linux.

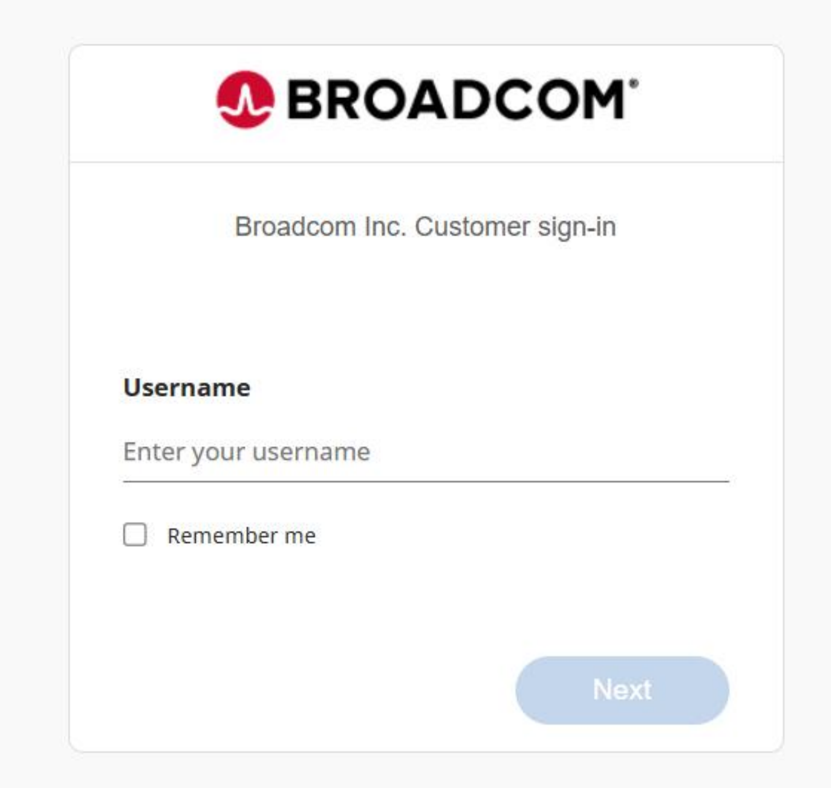
Server referinta

- Masina virtuala
- VMware Workstation Pro (Broadcom)



VMware Workstation Pro for PC

Build and test nearly any app with the world's leading desktop hypervisor app for Windows and Linux.



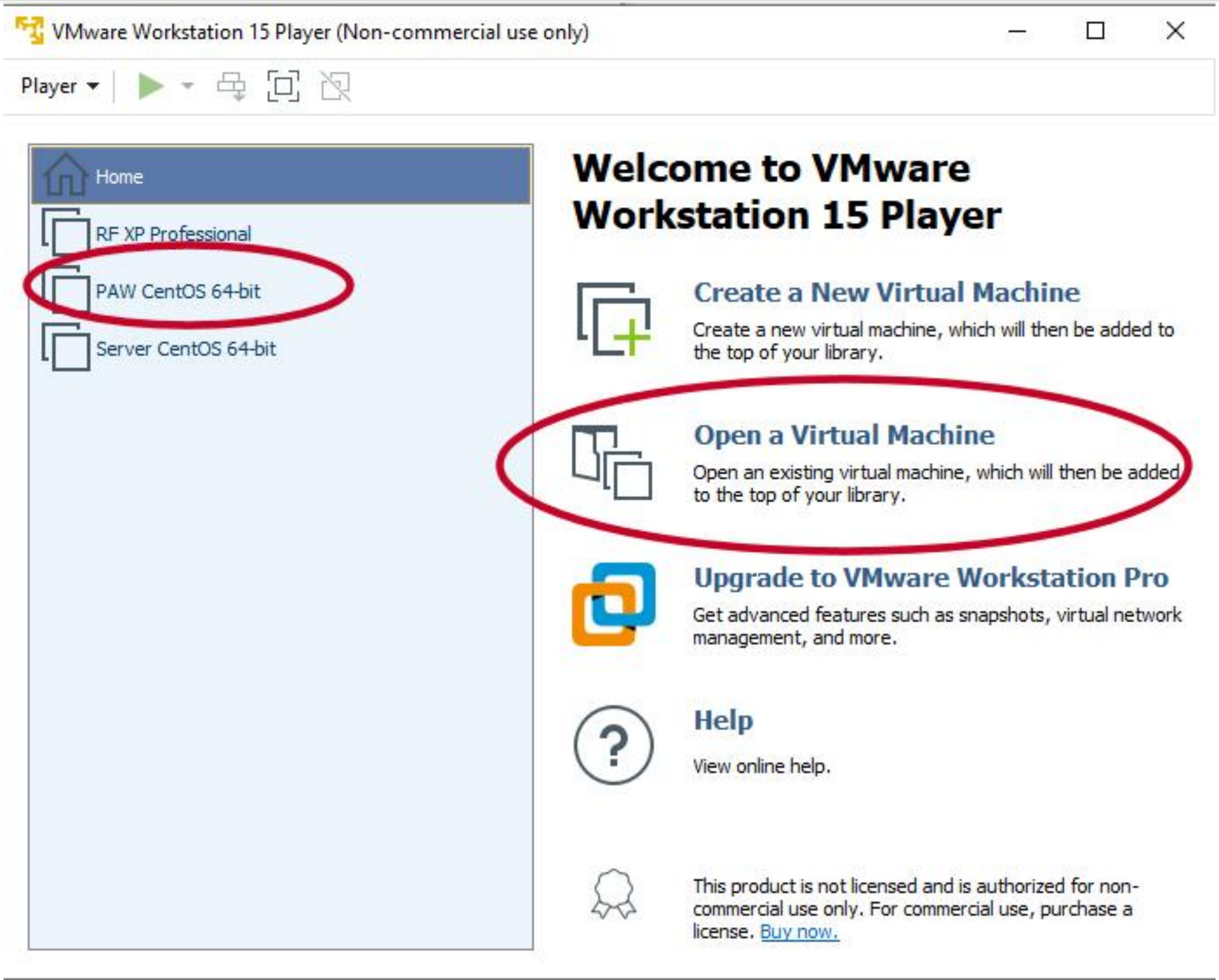
Broadcom Inc. Customer sign-in

Username

Enter your username

□ Remember me

Reference Server



VMware Workstation 15 Player (Non-commercial use only)

Home

RF XP Professional

PAW CentOS 64-bit

Server CentOS 64-bit

Welcome to VMware Workstation 15 Player

Create a New Virtual Machine

Create a new virtual machine, which will then be added to the top of your library.

Open a Virtual Machine

Open an existing virtual machine, which will then be added to the top of your library.

Upgrade to VMware Workstation Pro

Get advanced features such as snapshots, virtual network management, and more.

Help

View online help.

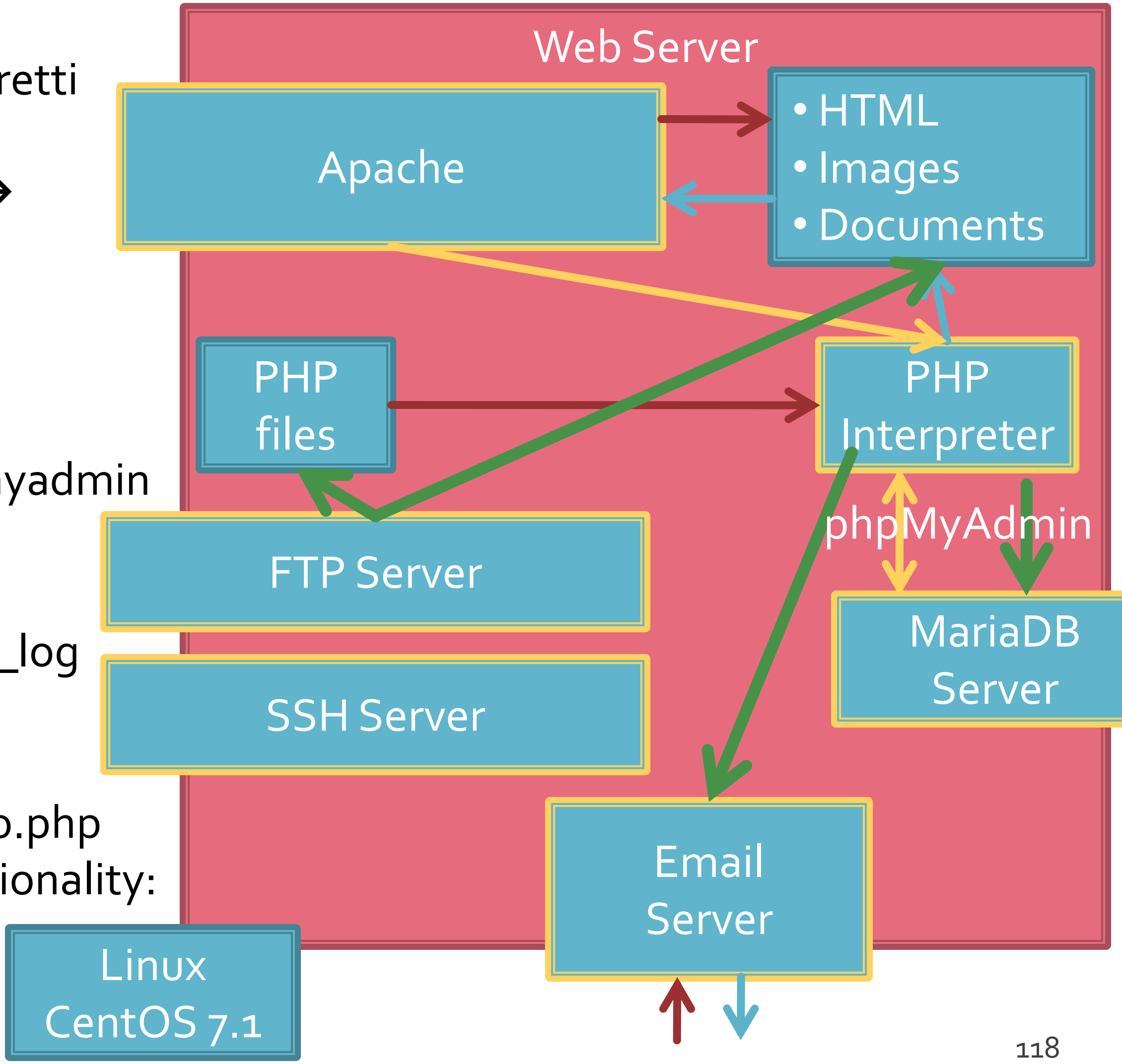
This product is not licensed and is authorized for non-commercial use only. For commercial use, purchase a license. Buy now.

Possible problems

- Current VMWare Player runs **only** on **64bit** operating systems Windows/Linux
  - for 32bit operating systems previous (**certified originals**) can be made available on rf-opto
- The host computer **must** enable **Hardware Virtualization**
  - Hardware Virtualization is enabled in BIOS, depending on the PC manufacturer: Processor, Chipset, Northbridge
  - Options name: VT-x, AMD-V, Vanderpool, Hyper-V, SVM, Intel Virtualization Technology. if available: Intel VT-d, AMD IOMMU
- VM archive requires **7zip** native to the target operating system

Using LAMP

1. login → root:masterrc / paw:masteretti
2. ifconfig → 192.168.30.5
3. putty.exe → 192.168.30.5 → SSH → root:masterrc (remote login)
4. [other linux command line]
5. FTP → Winscp → SFTP → student:masterrc@192.168.30.5
6. MySql → http://192.168.30.5/phpmyadmin → root:masterrc / 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
9. if DHCP service stops Apache functionality:
service httpd restart



LAMP Reference Server 2023

- Linux, two variants
  - Centos 7.1
    - PHP 5.4.16
    - MariaDB 5.5.44
    - Apache 2.4.6
    - **root**/student:masterrc
  - Ubuntu 20.04 (**recommended**)
    - PHP 7.4.3
    - MariaDB 10.3.31
    - Apache 2.4.41
    - **paw**/student:masteretti
    - correction **paw FTP access**:
      - sudo usermod -a -G upload paw
      - sudo chmod -R 775 /var/www

LAMP Reference Server 2025

- Linux, three proposed servers
  - CentOS 7.1
  - Ubuntu 20.04
  - **Debian 12.5**

LAMP Reference Server

- 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
  - creat: Workstation Player 12.x (**12**)

LAMP Reference Server

- 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
  - creat: Workstation Player 15.x (**16**)

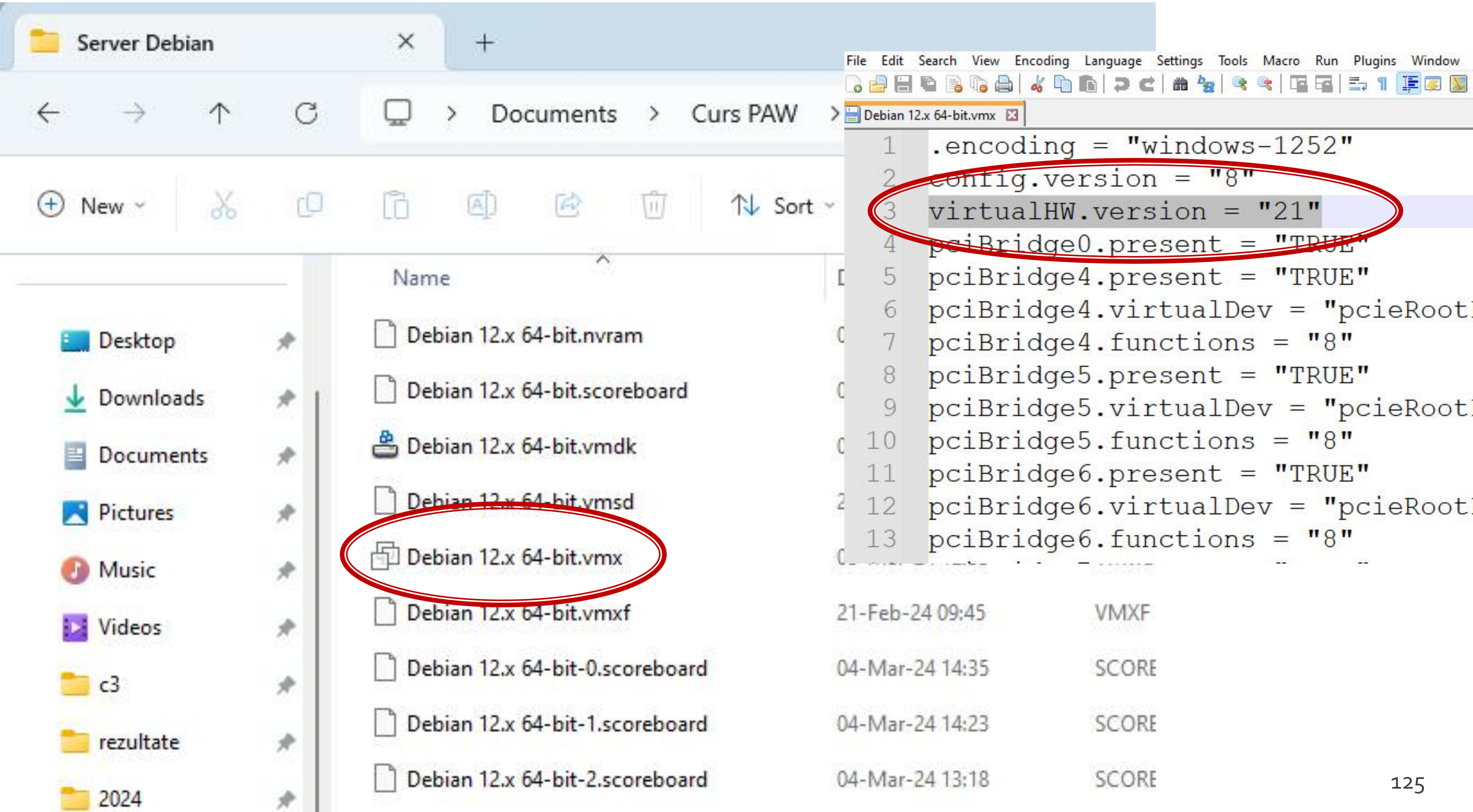
LAMP Reference Server

- 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
  - creat: Workstation Player 17.5 (**21**)

Server referinta

- Pentru rularea unui server pe o versiune VMware Player anterioara:
  - se localizeaza fisierul “*.vmx” a server-ului
  - se modifica virtualHW.version = “**21**” la o valoare mai mica (anterioara)
    - in 2.13 -> **18**

LAMP Reference Server



Name		
Debian 12.x 64-bit.nvram		
Debian 12.x 64-bit.scoreboard		
Debian 12.x 64-bit.vmdk		
Debian 12.x 64-bit.vmsd		
Debian 12.x 64-bit.vmx		
Debian 12.x 64-bit.vmxf	21-Feb-24 09:45	VMXF
Debian 12.x 64-bit-0.scoreboard	04-Mar-24 14:35	SCORE
Debian 12.x 64-bit-1.scoreboard	04-Mar-24 14:23	SCORE
Debian 12.x 64-bit-2.scoreboard	04-Mar-24 13:18	SCORE

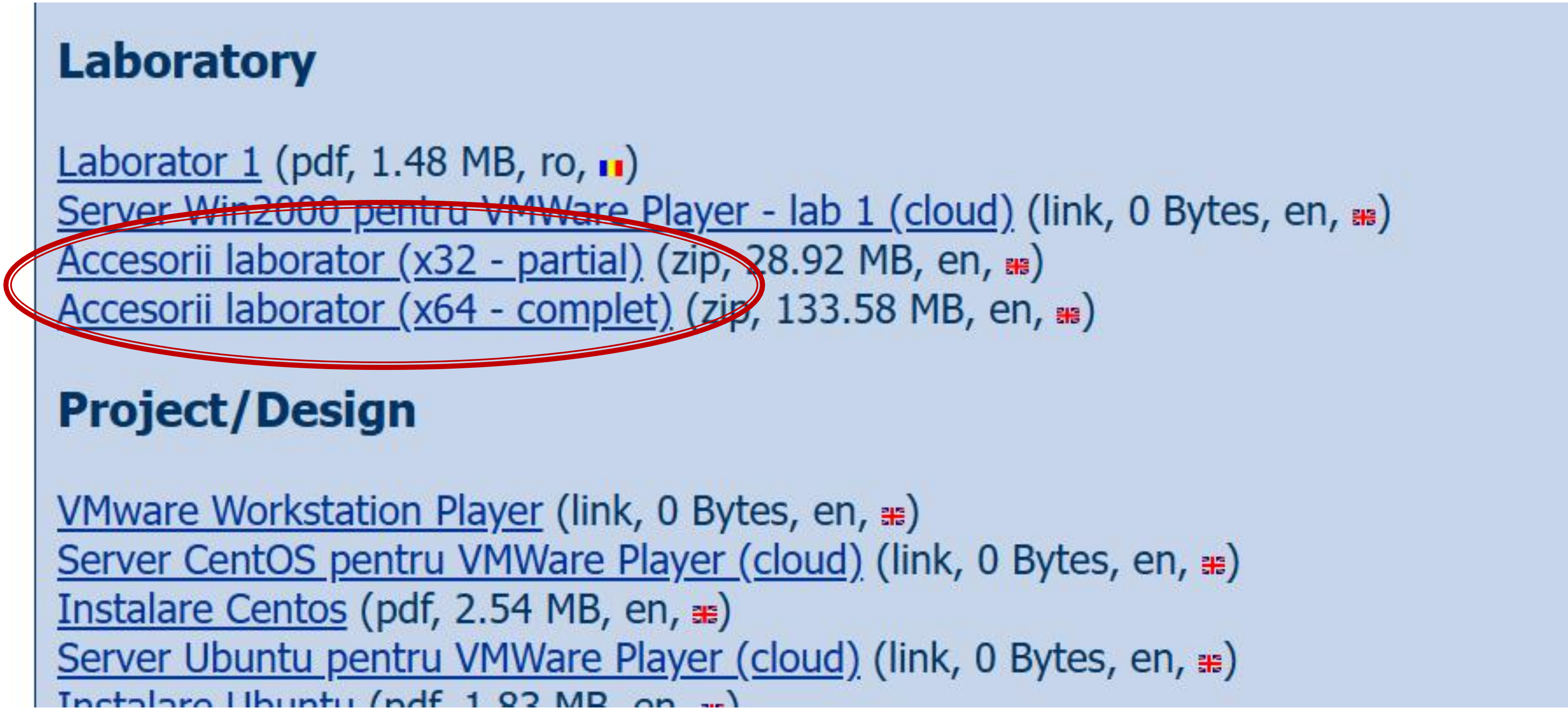
```
.encoding = "windows-1252"
config.version = "8"
virtualHW.version = "21"
pciBridge0.present = "TRUE"
pciBridge4.present = "TRUE"
pciBridge4.virtualDev = "pcieRoot
pciBridge4.functions = "8"
pciBridge5.present = "TRUE"
pciBridge5.virtualDev = "pcieRoot
pciBridge5.functions = "8"
pciBridge6.present = "TRUE"
pciBridge6.virtualDev = "pcieRoot
pciBridge6.functions = "8"
```

Support applications

- 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

Aplicatii suport

- Variante portabile



Laboratory

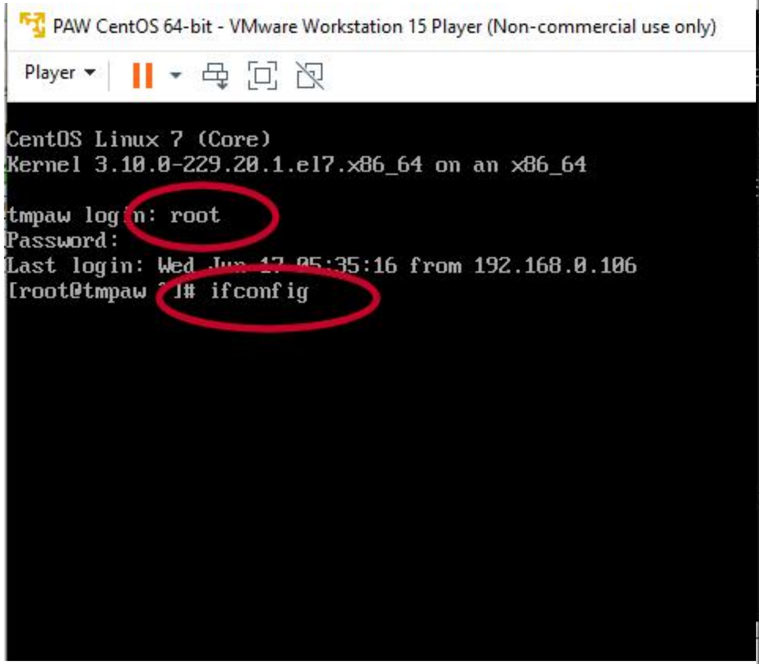
Laborator 1 (pdf, 1.48 MB, ro, 🇷🇴)
Server Win2000 pentru VMWare Player - lab 1 (cloud) (link, 0 Bytes, en, 🇬🇧)
Accesorii laborator (x32 - partial) (zip, 28.92 MB, en, 🇬🇧)
Accesorii laborator (x64 - complet) (zip, 133.58 MB, en, 🇬🇧)

Project/Design

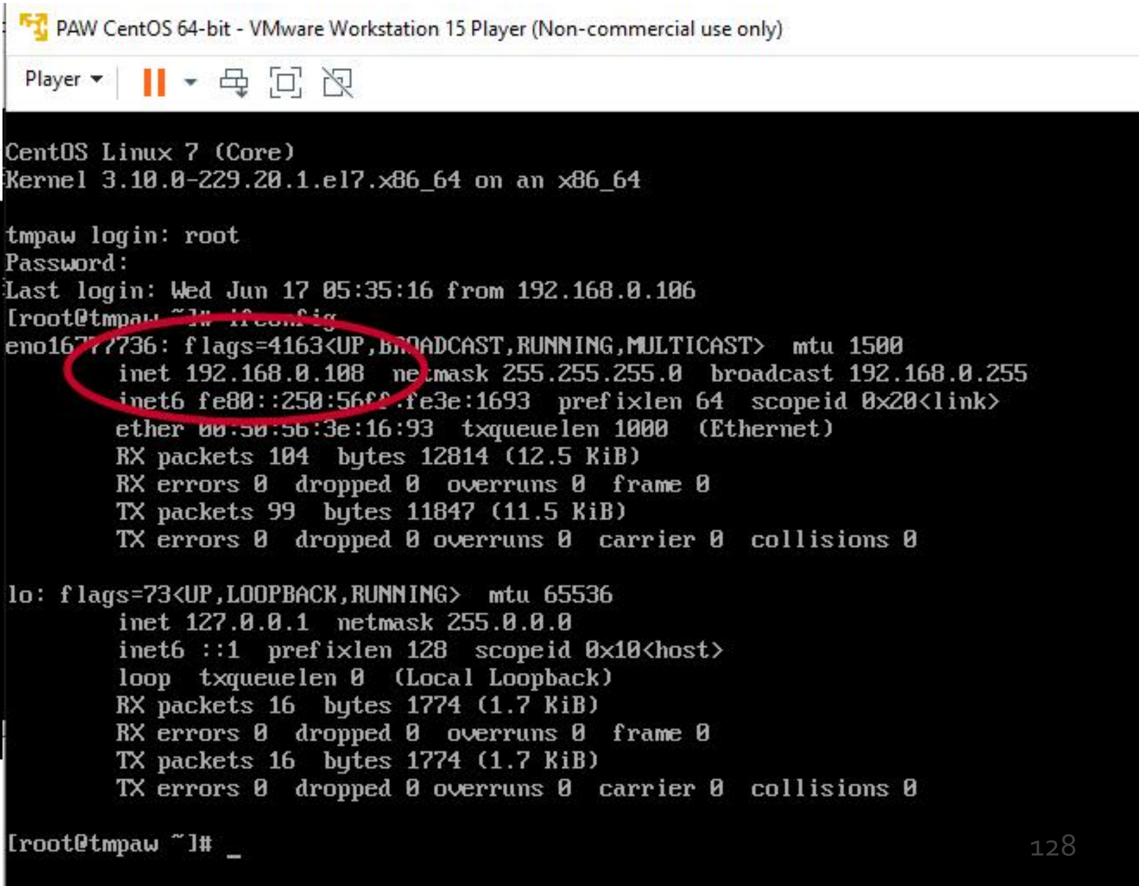
VMware Workstation Player (link, 0 Bytes, en, 🇬🇧)
Server CentOS pentru VMWare Player (cloud) (link, 0 Bytes, en, 🇬🇧)
Instalare Centos (pdf, 2.54 MB, en, 🇬🇧)
Server Ubuntu pentru VMWare Player (cloud) (link, 0 Bytes, en, 🇬🇧)
Instalare Ubuntu (pdf, 1.83 MB, en, 🇬🇧)

IP address

- login, ifconfig
- Ctrl + Alt + mouse


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

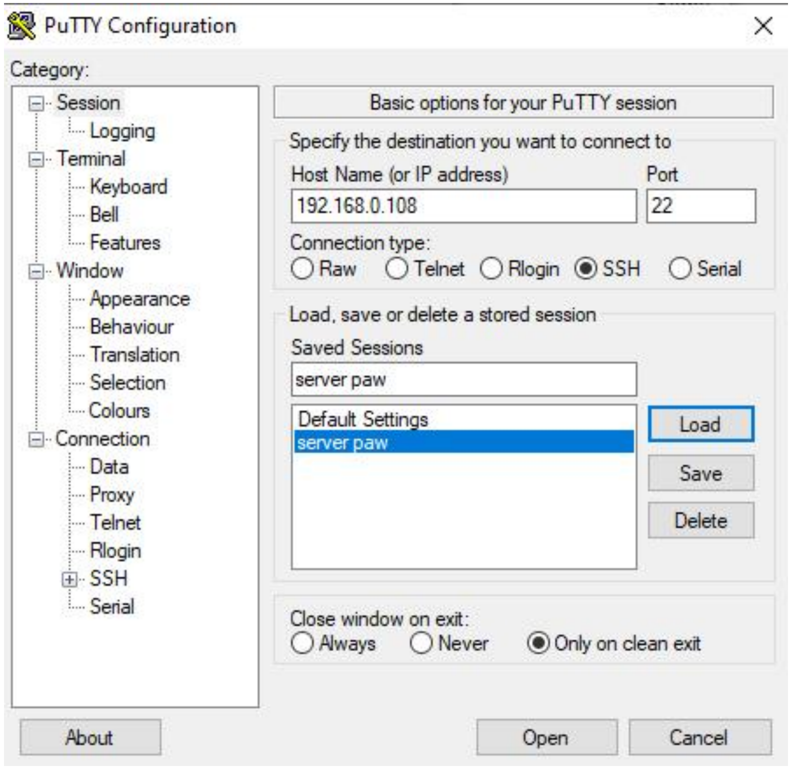
```
CentOS Linux 7 (Core)
Kernel 3.10.0-229.20.1.el7.x86_64 on an x86_64

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

```
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 00:50:56:3e: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
- avoids mouse capture (CentOS), copy/paste etc.



Specify the destination you want to connect to

Host Name (or IP address): 192.168.0.108

Port: 22

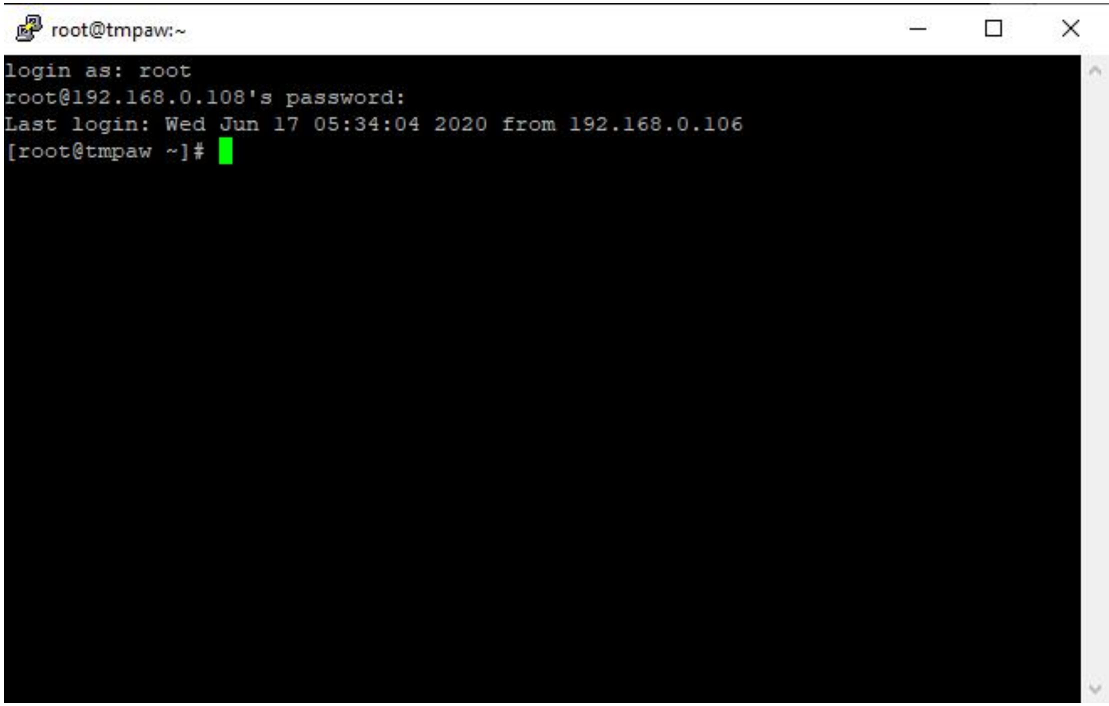
Connection type: ○ Raw ○ Telnet ○ Rlogin ● SSH ○ Serial

Load, save or delete a stored session

Saved Sessions: server paw

Default Settings
server paw

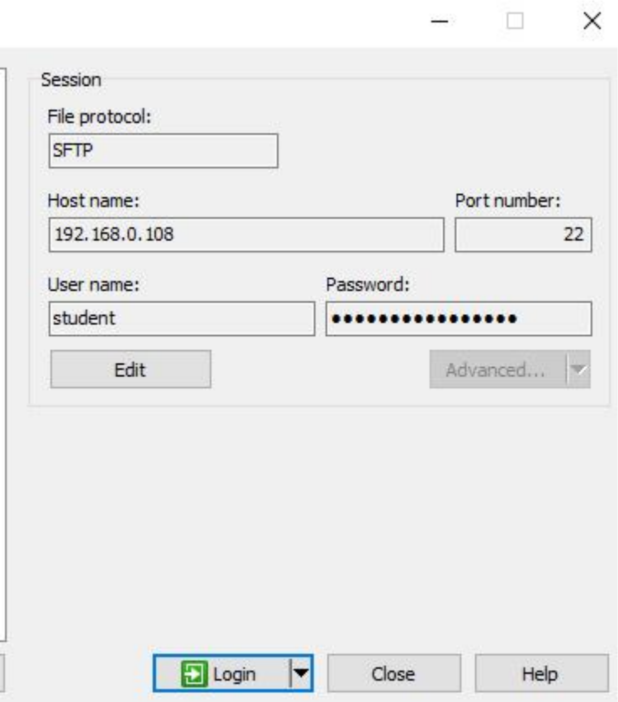
Close window on exit: ○ Always ○ Never ● Only on clean exit



```
login as: root
root@192.168.0.108's password:
Last login: Wed Jun 17 05:34:04 2020 from 192.168.0.106
[root@tmpaw ~]#
```

WinSCP

- FTP client
- upload files



Session

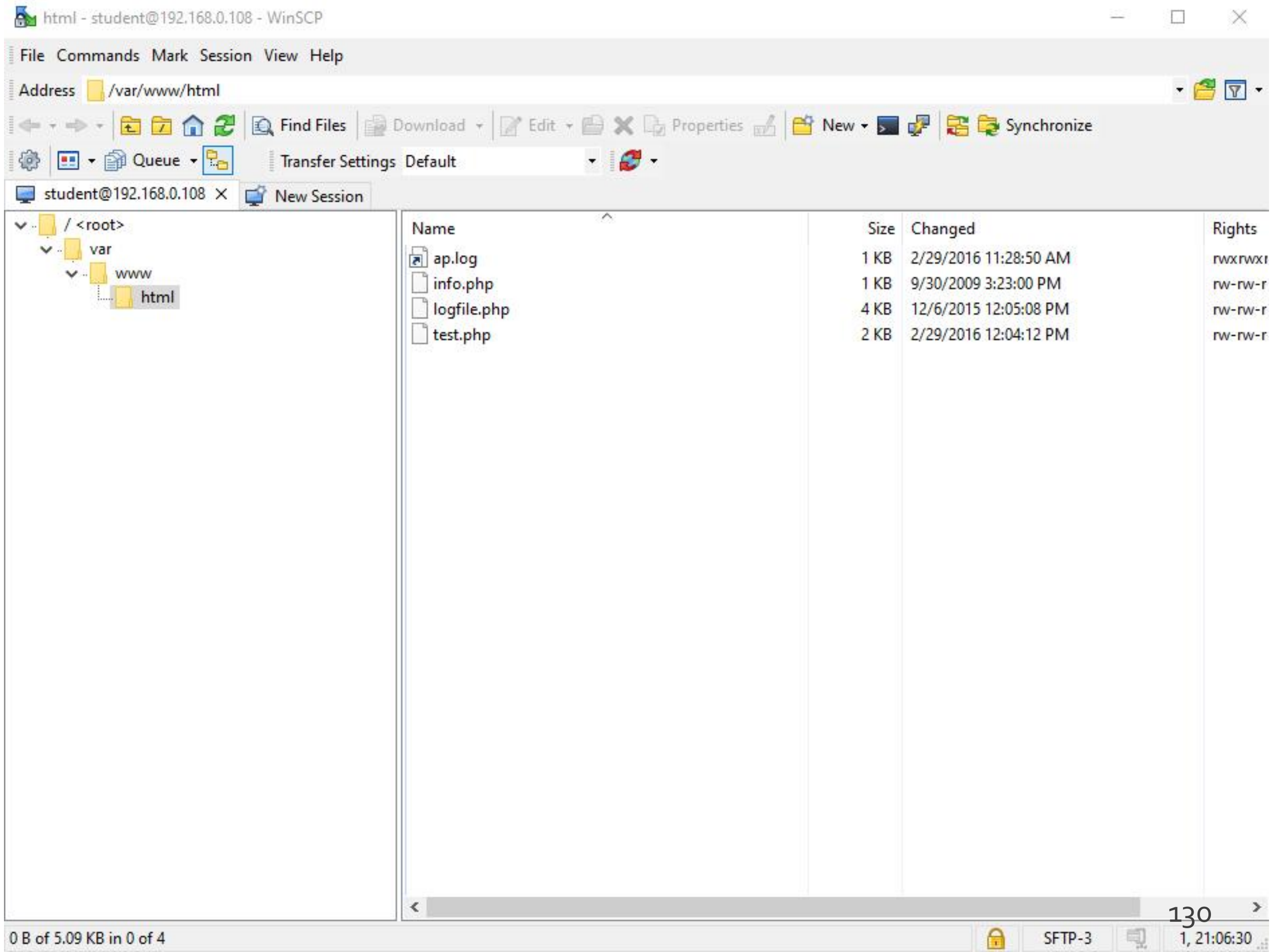
File protocol: SFTP

Host name: 192.168.0.108

Port number: 22

User name: student

Password: ●●●●●●●●●●●●●●●●●



html - student@192.168.0.108 - WinSCP

Address /var/www/html

Transfer Settings Default

student@192.168.0.108

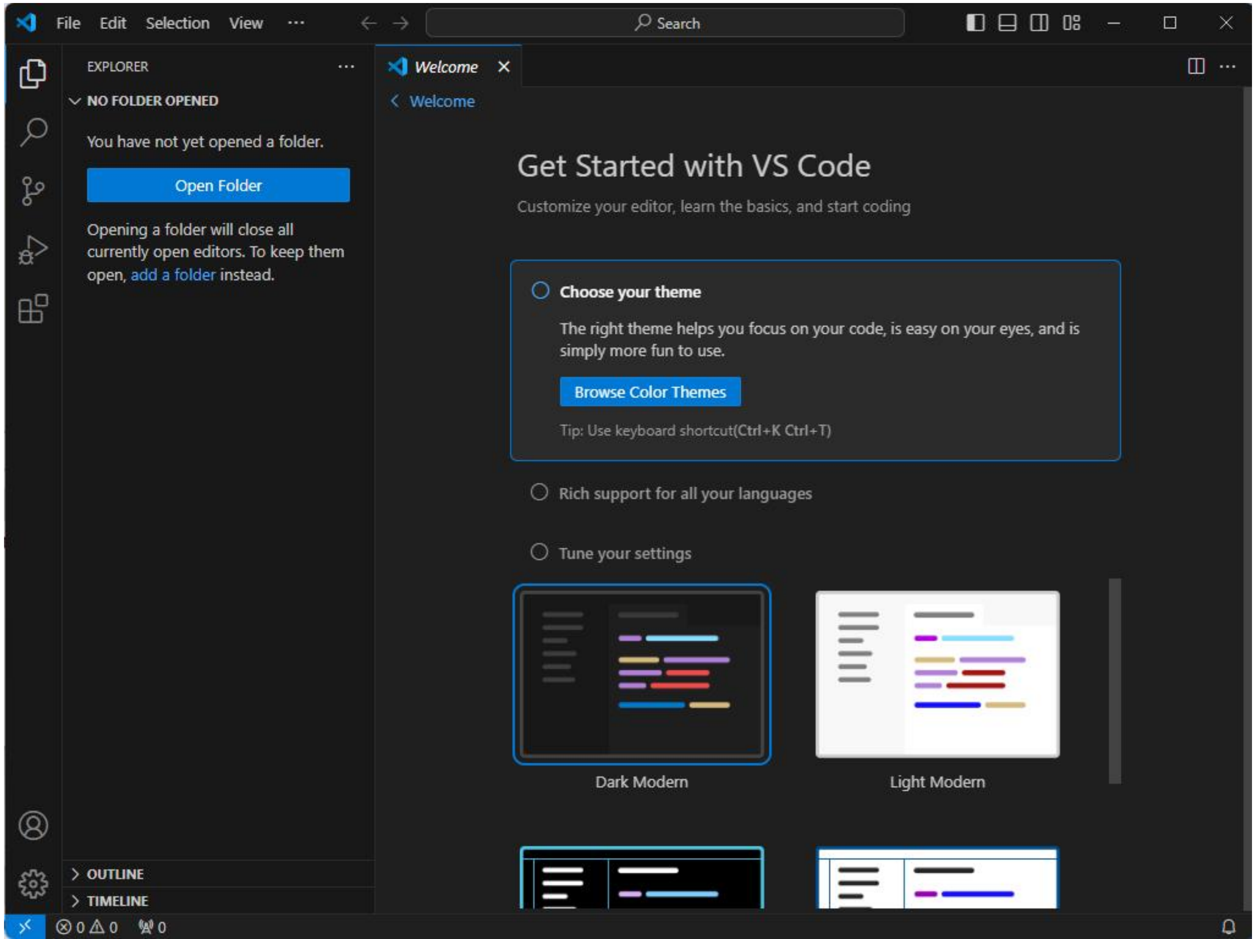
Name	Size	Changed	Rights
ap.log	1 KB	2/29/2016 11:28:50 AM	rwxrwxr
info.php	1 KB	9/30/2009 3:23:00 PM	rw-rw-r
logfile.php	4 KB	12/6/2015 12:05:08 PM	rw-rw-r
test.php	2 KB	2/29/2016 12:04:12 PM	rw-rw-r

0 B of 5.09 KB in 0 of 4

1, 21:06:30

Visual Studio Code

- 1.87 Portable



You have not yet opened a folder.

Opening a folder will close all currently open editors. To keep them open, add a folder instead.

Get Started with VS Code

Customize your editor, learn the basics, and start coding

○ Choose your theme

The right theme helps you focus on your code, is easy on your eyes, and is simply more fun to use.

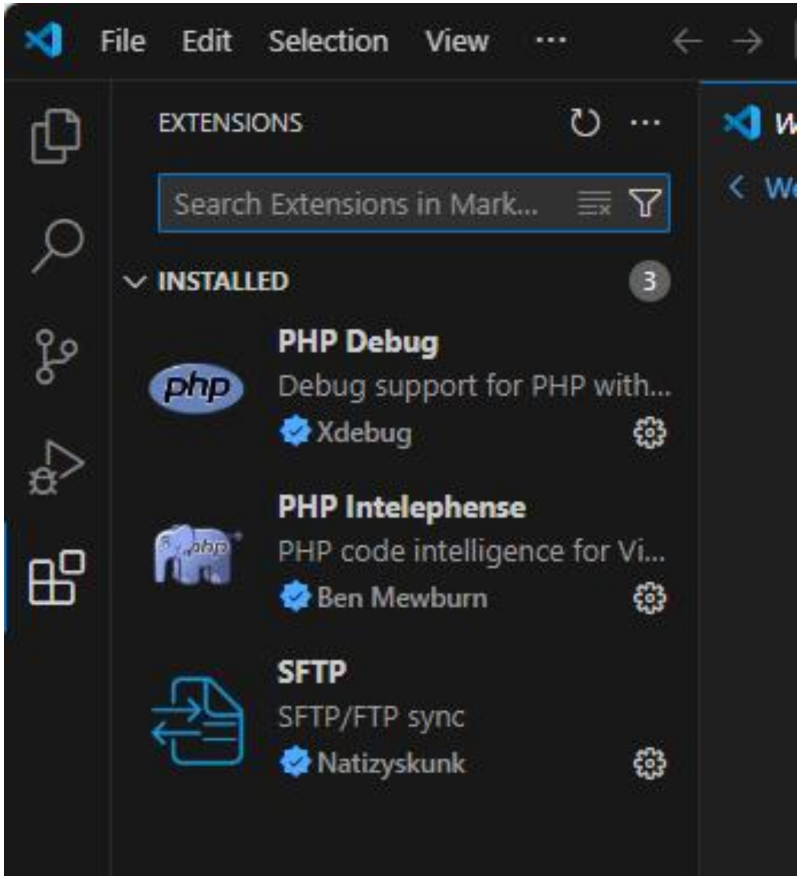
Tip: Use keyboard shortcut(Ctrl+K Ctrl+T)

○ Rich support for all your languages

○ Tune your settings

Visual Studio Code

- Installed extensions
  - PHP Intelephense
    - PHP 8 -> Debian
  - PHP Debug (inactive for now)
  - SFTP – automatic save on server

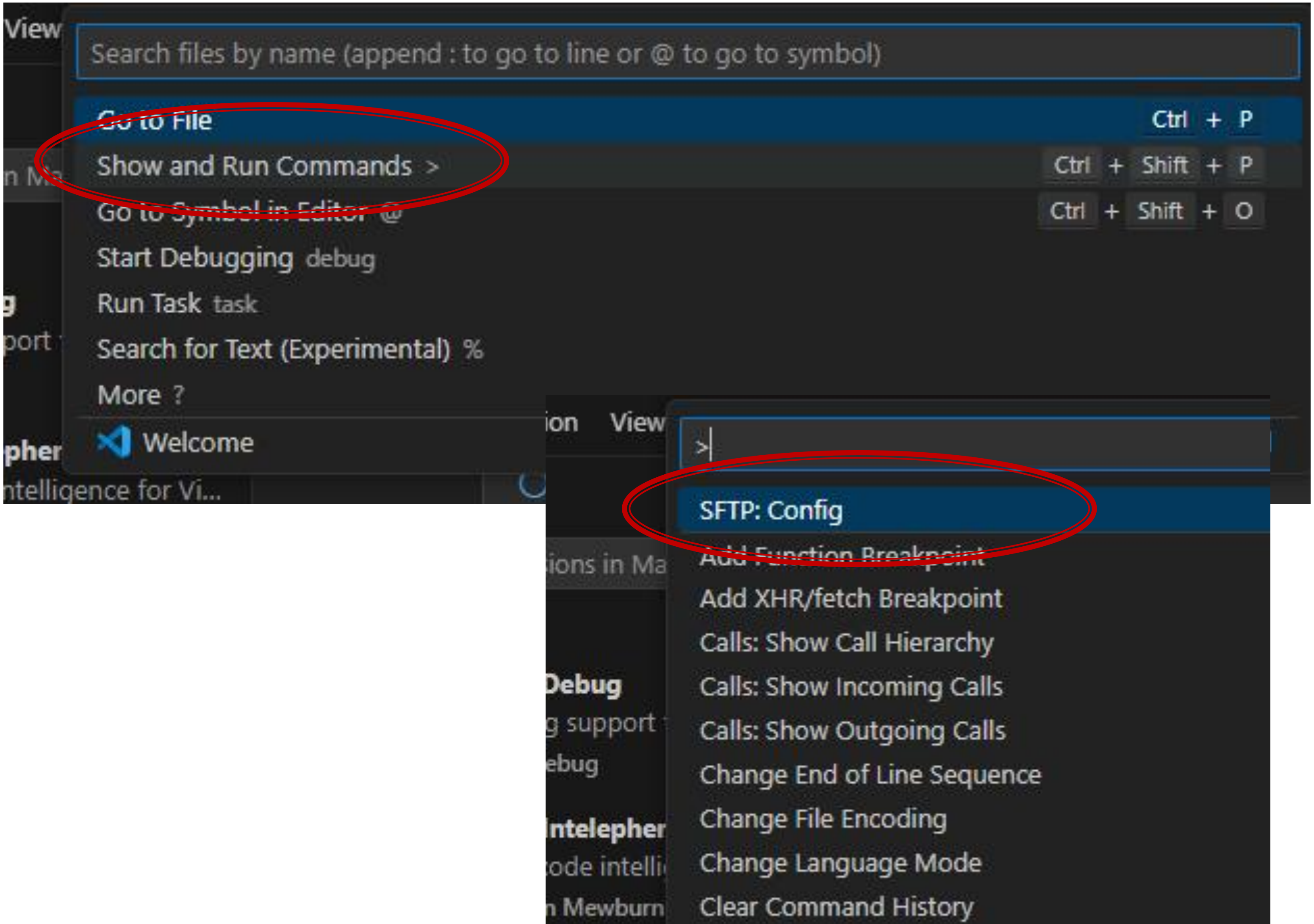


PHP Debug
Debug support for PHP with...
Xdebug

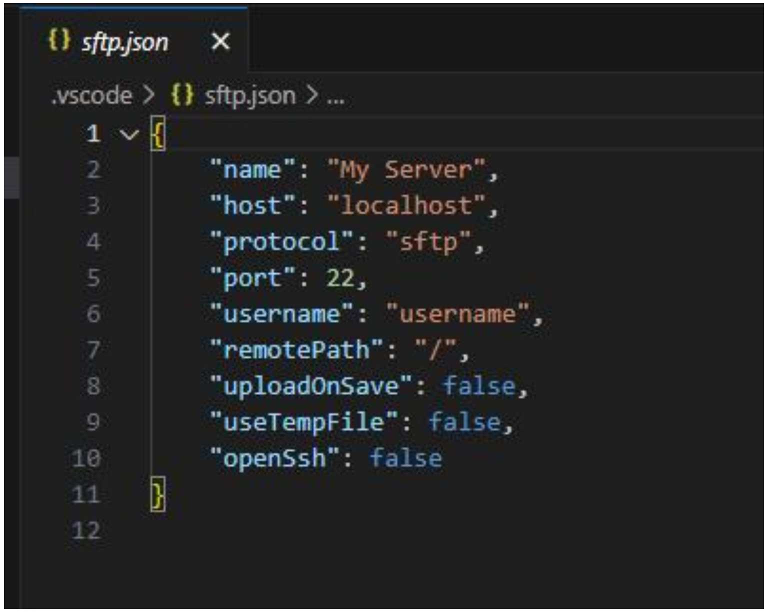
PHP Intelephense
PHP code intelligence for Vi...
Ben Mewburn

SFTP
SFTP/FTP sync
Natizyskunk

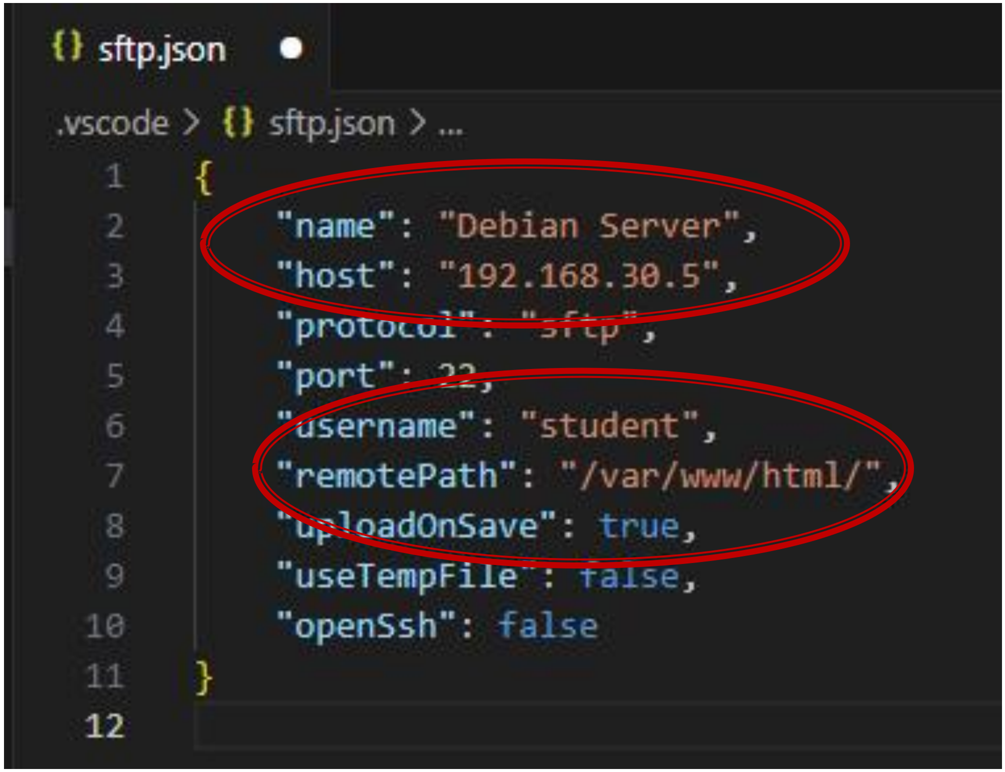
Visual Studio Code



Visual Studio Code

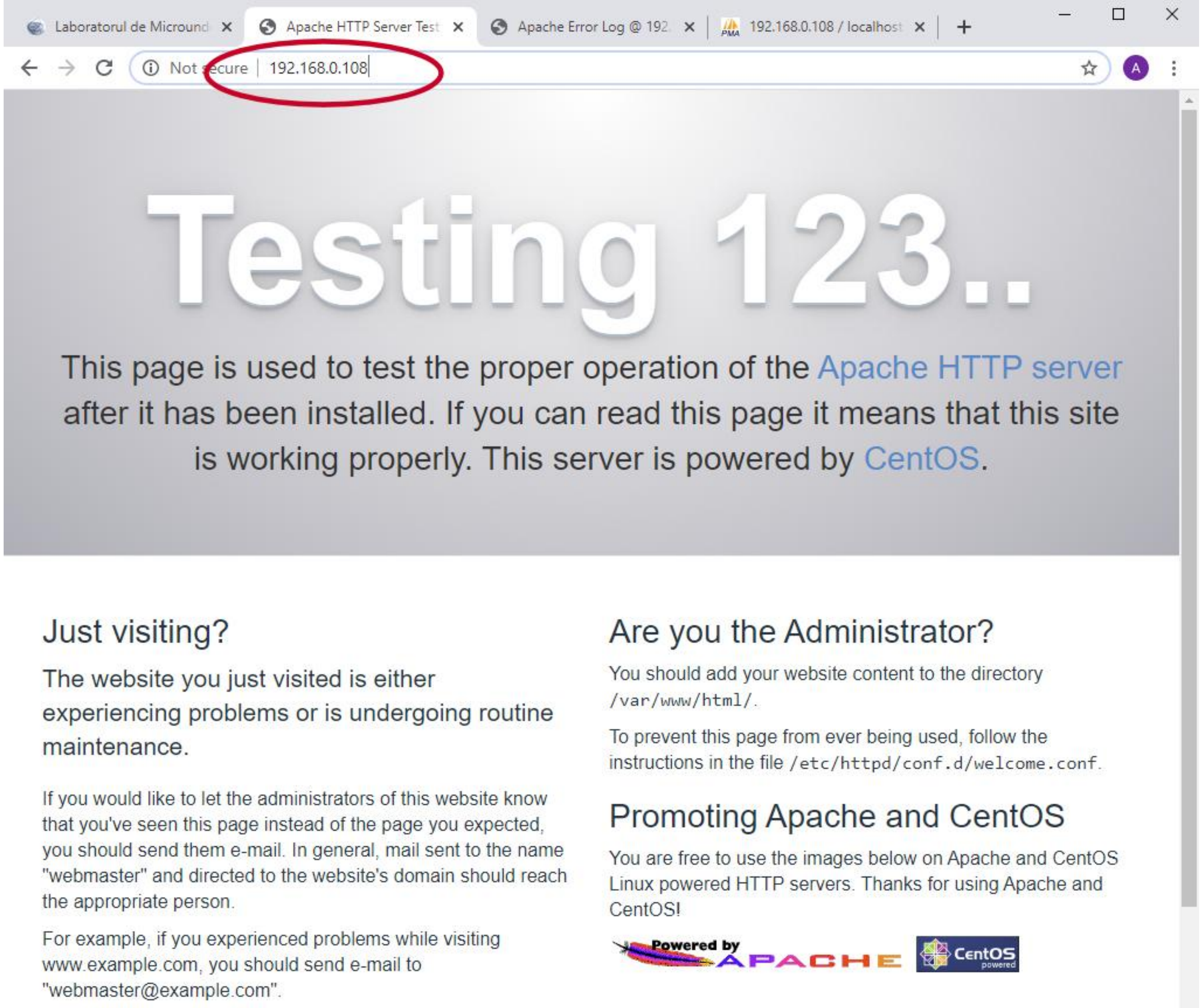


```
{
    "name": "My Server",
    "host": "localhost",
    "protocol": "sftp",
    "port": 22,
    "username": "username",
    "remotePath": "/",
    "uploadOnSave": false,
    "useTempFile": false,
    "openSsh": false
}
```



```
{
    "name": "Debian Server",
    "host": "192.168.30.5",
    "protocol": "sftp",
    "port": 22,
    "username": "student",
    "remotePath": "/var/www/html/",
    "uploadOnSave": true,
    "useTempFile": false,
    "openSsh": false
}
```

Browser



Testing 123..

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page it means that this site is working properly. This server is powered by CentOS.

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?

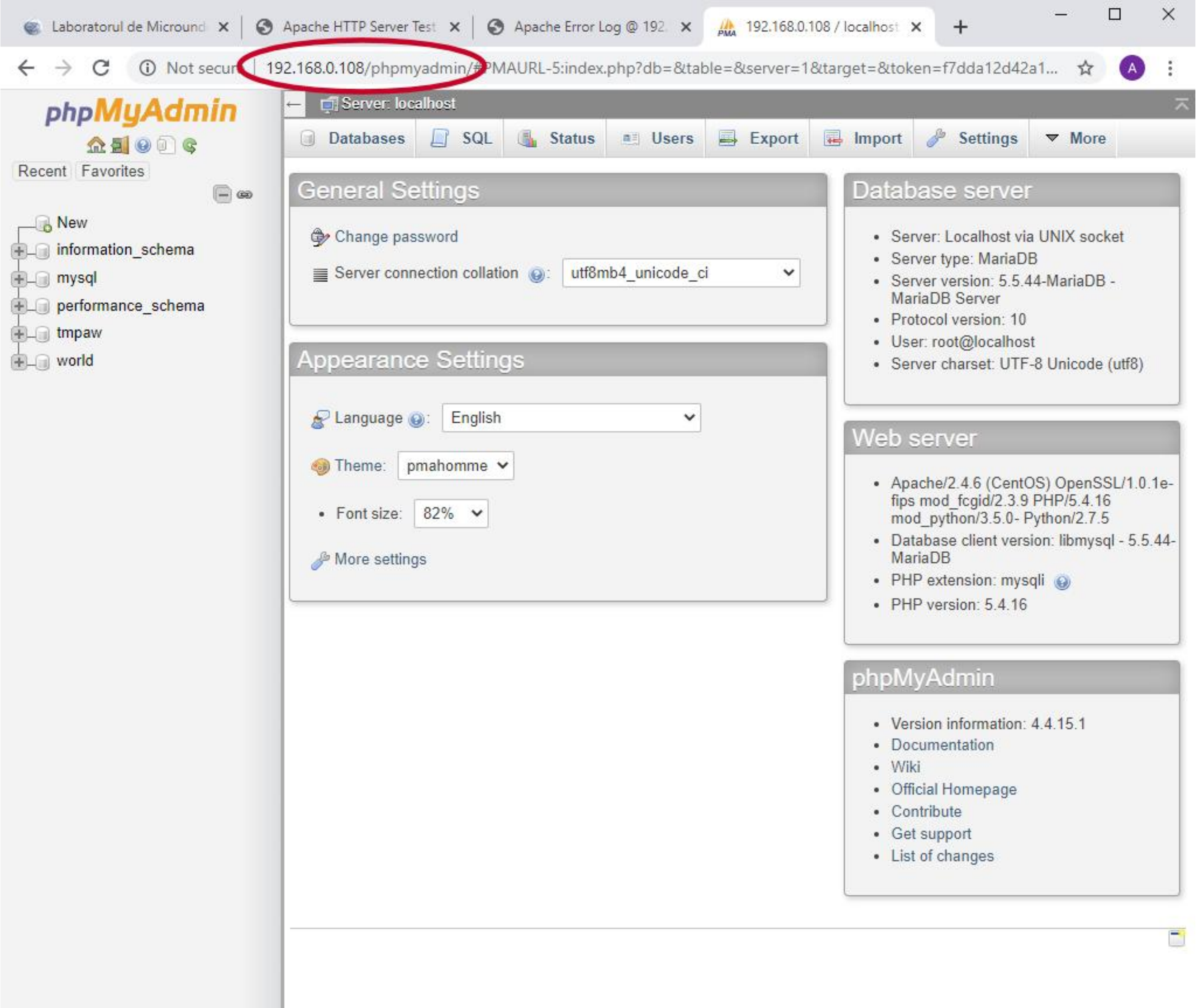
You should add your website content to the directory `/var/www/html/`.

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



192.168.0.108/phpmyadmin/#PMAURL-5:index.php?db=&table=&server=1&target=&token=f7dda12d42a1...

Server: localhost

New
information_schema
mysql
performance_schema
tmpaw
world

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: mysqli
- 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
- https://rf-opto.etti.tuiasi.ro
- rdamian@etti.tuiasi.ro