**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**

Інститут **КНІТ**

Кафедра **ПЗ**

ЗВІТ

До лабораторної роботи № 5

**З дисципліни:** *“* *РНР як засіб написання сценаріїв.”*

**На тему:** *“* *Використання React JS та Bootstrap”*

**Лектор:**

проф. каф. ПЗ

Мельник Р. А.

**Виконав:**

ст. гр. ПЗ-23 Михалевич П.-І. В.

**Прийняв:**

доц. каф. ПЗ

Грицай О.Д.

« » 2022 р.

∑= .

Львів – 2022

**Тема роботи:** РНР як засіб написання сценаріїв.

**Мета роботи:** Засвоїти елементи програмування серверної частини застосувань мовою РНР.

**ТЕОРЕТИЧНІ ВІДОМОСТІ**

PHP – це мова програмування, яка була розроблена для генерації HTML сторінок на стороні сервера. Цю мову використовують для так званих динамічних сторінок, тому, що вона дає можливість генерувати HTML код, подібним чином, як це можна робити у JSX React.

На даний момент цю мову використовують переважно для обробки запитів користувачів зі сторони сервера. Однією із головних переваг PHP є те, що її код може бути безпосередньо вбудованим в HTML-код сторінки.

PHP є ні компілятором, ані інтерпретатором, він обробляє сценарії, які подаються на вхід, потім він транслює їх в спеціальний байт-код, який і виконує. У такому разі виконавчий файл не створюється. Також цей код є значно компактнішим, аніж код програми, тому він виконується швидше.

**ЗАВДАННЯ**

Встановити та налаштувати PHP і локальний сервер.

Написати серверну частину сайту, яка буде приймати дані гравця в шахи для збереження (фактичне зберігання даних поки не потрібно) валідувати їх та відповідати “Дані успішно збережено” або повертати інформацію про невалідні дані.

**ХІД РОБОТИ**

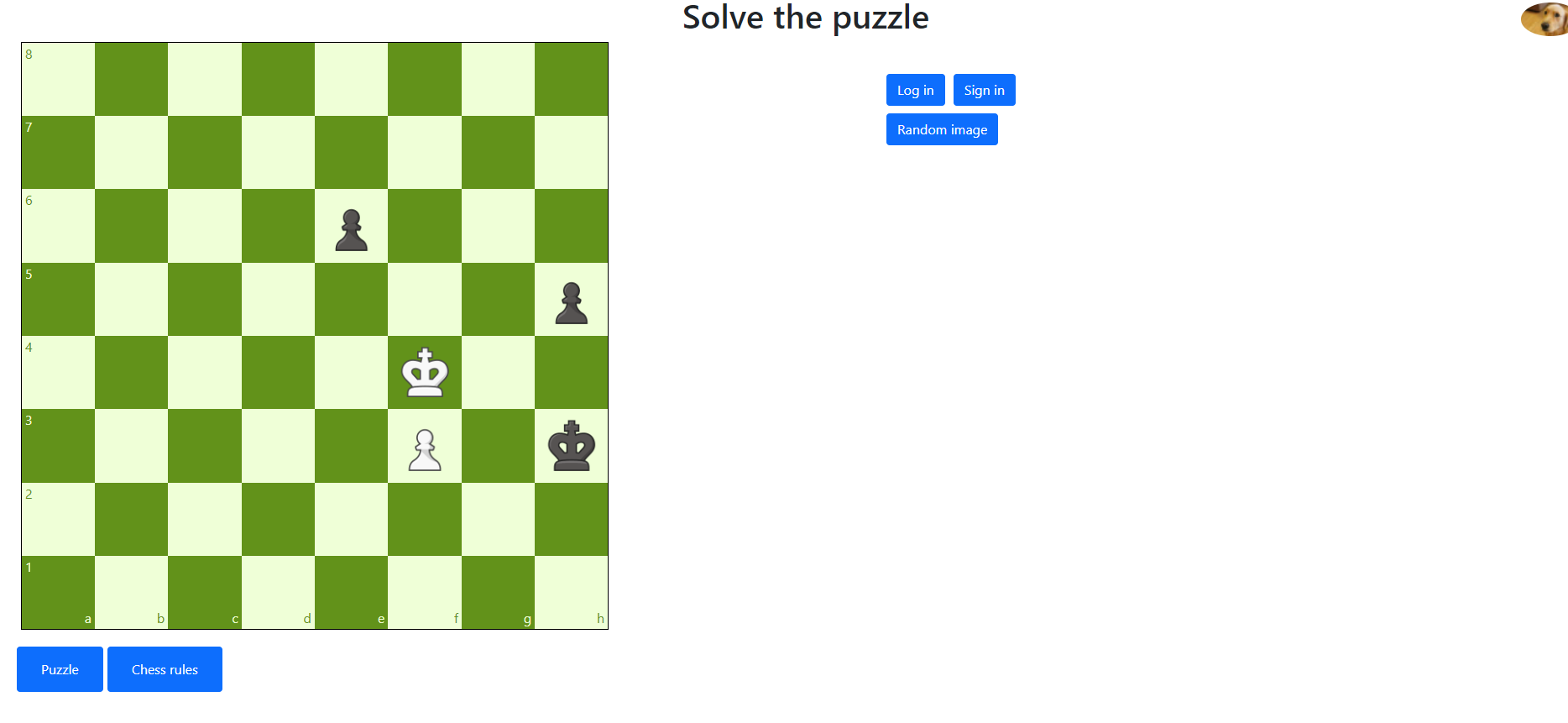
****

Рис.1. Початкова сторінка

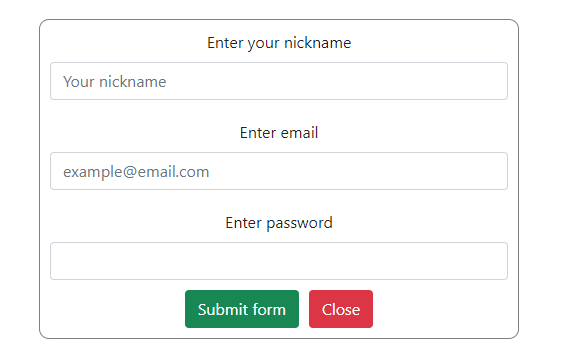


Рис.2. Форма входу для гравця

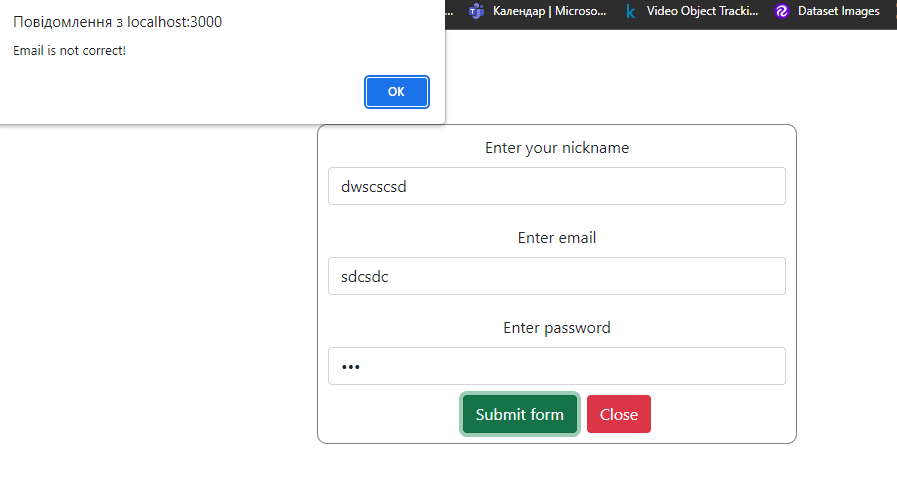


Рис.3. Валідація форми

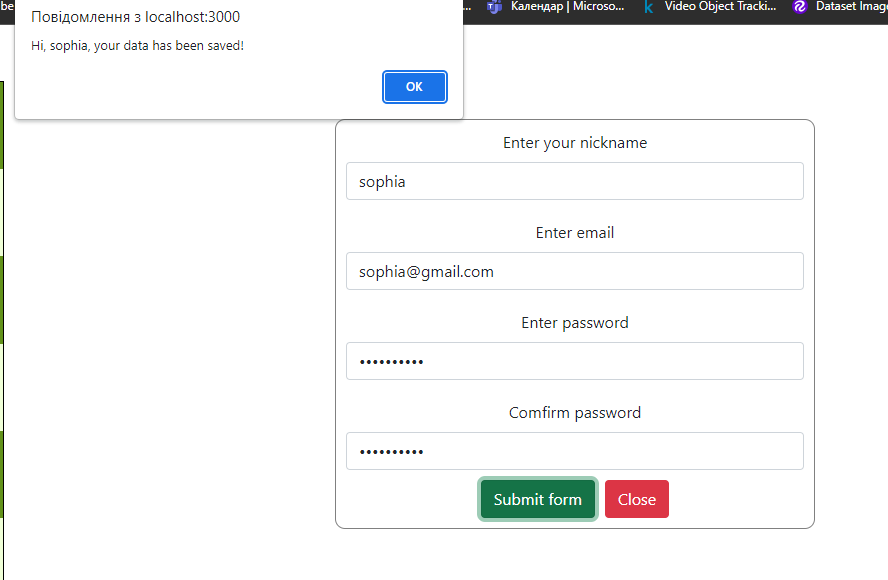


Рис. 4. Успішне збереження даних у БД

Код програми:

Login.php:

<?php

header('Access-Control-Allow-Origin: \*');

$servername = "localhost";

$username = "root";

$password = "root";

$dbname = "chess\_players";

// Create connection

$conn = mysqli\_connect($servername, $username, $password, $dbname);

$response = array('sever\_error' => false,

                  'server\_error\_text' => "",

                  'validated'=>true,

                  'validation\_error\_text' => '',

                  'nickname' => '');

// Check connection

if (!$conn) {

  $response['sever\_error'] = true;

  $response['server\_error\_text'] = "Connection failed: " . mysqli\_connect\_error();

}

$nickname = $\_POST['nickname'];

$email = $\_POST['email'];

$pwd = $\_POST['password'];

//Validation

if(empty($nickname) or empty($email) or empty($pwd)){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Not all data was provided!";

}

if(!filter\_var($email, FILTER\_VALIDATE\_EMAIL)  and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Email is not correct! ";

}

if(!preg\_match("/^(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z])[0-9a-zA-Z]{8,}$/",$pwd)  and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Password must have at least 1 upper and lower case letter, 1 number and consist of 8 characters! ";

}

if(strlen($nickname) < 3 and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Nickname must be at least 3 characters! ";

}

$db\_answer = mysqli\_query($conn, "SELECT \* FROM chess WHERE (username = '$nickname')");

if (db\_answer) {

  $user = mysqli\_fetch\_assoc($db\_answer);

  if($user != null){

    if($pwd != $user['password']){

      if($response['validated'] != false){

        $response['validated'] = false;

        $response['validation\_error\_text'] = "Invalid password! ";

      }

    }else{

      $response['nickname'] = $nickname;

    }

  }else if($response['validated'] != false){

    $response['validated'] = false;

    $response['validation\_error\_text'] = "No such nickname registered! ";

  }

} else if($response['sever\_error'] != true){

  $response['sever\_error'] = true;

  $response['server\_error\_text'] = "DB error! ";

}

$conn->close();

echo json\_encode($response);

?>

Registration.php:

<?php

header('Access-Control-Allow-Origin: \*');

$servername = "localhost";

$username = "root";

$password = "root";

$dbname = "chess\_players";

// Create connection

$conn = mysqli\_connect($servername, $username, $password, $dbname);

$response = array('sever\_error' => false,

                  'server\_error\_text' => "",

                  'validated'=> true,

                  'validation\_error\_text' => '',

                  'nickname' => '');

// Check connection

if (!$conn) {

  $response['sever\_error'] = true;

  $response['server\_error\_text'] = "Connection failed: " . mysqli\_connect\_error();

}

$nickname = $\_POST['nickname'];

$email = $\_POST['email'];

$pwd = $\_POST['password'];

//Validation

if(empty($nickname) or empty($email) or empty($pwd)){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Not all data was provided!";

}

if(!filter\_var($email, FILTER\_VALIDATE\_EMAIL) and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Email is not correct! ";

}

if(!preg\_match("/^(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z])[0-9a-zA-Z]{8,}$/",$pwd) and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Password must have at least 1 upper and lower case letter, 1 number and consist of 8 characters! ";

}

if(strlen($nickname) < 3 and $response['validated'] != false){

  $response['validated'] = false;

  $response['validation\_error\_text'] = "Nickname must be at least 3 characters! ";

}

$db\_answer = mysqli\_query($conn, "SELECT \* FROM chess WHERE (username = '$nickname')");

if ($db\_answer) {

  $user = mysqli\_fetch\_assoc($db\_answer);

  if($user != null){

    if($response['validated'] != false){

      $response['validated'] = false;

      $response['validation\_error\_text'] = "Such nickname already exist! ";

    }

  }else{

    $result = mysqli\_query($conn, "INSERT INTO chess (username, email, password) VALUES ('$nickname', '$email', '$pwd')");

    if ($result) {

      $response['nickname'] = $nickname;

    } else {

      if($response['sever\_error'] != true){

        $response['sever\_error'] = true;

        $response['server\_error\_text'] = "DB error! Can't insert! ";

      }

    }

  }

} else {

  if($response['sever\_error'] != true){

    $response['sever\_error'] = true;

    $response['server\_error\_text'] = "DB error! Can't get data! ";

  }

}

$conn->close();

echo json\_encode($response);

?>

**Висновок**

На даній лабораторній роботі було отримано базові знання про будову вебсайтів. Досліджено поняття доменів та серверів. Отримано базові знання синтаксису мови PHP. Встановлено, що програми написані цією мовою використовуються на серверах для оброблення запитів клієнтів. Встановлено Open Server. З його допомогою створено локальний сервер. З використанням якого, поєднано React JS та PHP. Написано серверну частину, яка валідує інформацію отриману від користувача.