

Лабораторна робота №6

Тема: Асинхронні запити на сервер та їх обробка

Мета роботи: ознайомитися з асинхронними запитами на сервер та їх обробкою.

Хід роботи:

Завдання 1: Система керування користувачами

Опис завдання: Створіть веб-додаток для системи керування користувачами, який використовує асинхронні запити для взаємодії між клієнтом та сервером. У цьому завданні передбачено дві основні функціональності: реєстрація нового користувача та перегляд списку користувачів.

1. Реєстрація користувача (фронтенд):

- Створіть HTML-форму для реєстрації нового користувача з полями для ім'я, електронної адреси та пароля.
- Використовуйте JavaScript для збору даних з форми та відправки їх на сервер за допомогою fetch API.
- Перевірте, чи всі поля заповнені перед відправкою на сервер.

2. Реєстрація користувача (бекенд):

- Створіть PHP-скрипт для обробки запиту на реєстрацію нового користувача.
- Перевірте правильність введених даних, такі як унікальність електронної адреси та мінімальна довжина пароля.
- Збережіть дані нового користувача в базі даних MySQL

3. Отримання списку користувачів (фронтенд):

- Створіть кнопку або посилання для відображення списку користувачів.
- Використовуйте JavaScript для відправки запиту на сервер для отримання списку користувачів.

					Житомирська політехніка.24.121.09.000 – Лр6			
Змн.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Степанець Р. С.			Звіт з лабораторної роботи		Літ.	Арк.
Перевір.		Ковтун В.В.						1
Керівник								22
Н. контр.							ФІКТ Гр. ВТк-24-1	
Зав. каф.								

- Виведіть отриманий список користувачів на веб-сторінці.

4. Отримання списку користувачів (бекенд):

- Створіть PHP-скрипт, який вибирає всіх користувачів з бази даних.
- Поверніть список користувачів у форматі JSON, щоб його можна було легко обробити на клієнті.
- Додайте можливість входу користувача за допомогою електронної адреси та пароля.
- Реалізуйте можливість видалення користувача.
- Додайте можливість редагування даних користувача.

Лістинг:

```
<?php
```

```
$data = json_decode(file_get_contents("php://input"));
```

```
$conn = new mysqli("localhost", "root", "", "lab6");
```

```
$id = (int)$data->id;
```

```
$conn->query("DELETE FROM users WHERE id=$id");
```

```
echo "Користувача видалено.";
```

```
?>
```

```
<?php
```

```
$conn = new mysqli("localhost", "root", "", "lab6");
```

```
$res = $conn->query("SELECT id, name, email FROM users");
```

```
$users = array();
```

```
while($row = $res->fetch_assoc()) {
```

```
    $users[] = $row;
```

```
}
```

```
echo json_encode($users);
```

```
?>
```

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		2

```

<?php
$data = json_decode(file_get_contents("php://input"));
$conn = new mysqli("localhost", "root", "", "lab6");
$email = $conn->real_escape_string($data->email);
$password = $data->password;

$res = $conn->query("SELECT * FROM users WHERE email='$email'");
if ($res->num_rows == 1) {
    $user = $res->fetch_assoc();
    if (password_verify($password, $user['password'])) {
        echo json_encode(["status" => "success", "user_id" => $user['id']]);
    } else {
        echo json_encode(["status" => "error", "message" => "Невірний пароль"]);
    }
} else {
    echo json_encode(["status" => "error", "message" => "Користувача не знайдено"]);
}
?>

```

```

<?php
$data = json_decode(file_get_contents("php://input"));
$conn = new mysqli("localhost", "root", "", "lab6");
$email = $conn->real_escape_string($data->email);
$check = $conn->query("SELECT * FROM users WHERE email='$email'");
if ($check->num_rows > 0) {
    echo "Користувач з таким email вже існує.";
    exit;
}

$name = $conn->real_escape_string($data->name);

```

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		3

```
$password = password_hash($data->password, PASSWORD_BCRYPT);
$conn->query("INSERT INTO users (name, email, password) VALUES ('$name',
'$email', '$password')");
echo "Користувача зареєстровано!";
?>
```

```
body {
    font-family: sans-serif;
    margin: 20px;
}
input, button {
    margin: 5px;
    padding: 8px;
}
```

```
<?php
$data = json_decode(file_get_contents("php://input"));
$conn = new mysqli("localhost", "root", "", "lab6");
$id = (int)$data->id;
$name = $conn->real_escape_string($data->name);
$email = $conn->real_escape_string($data->email);
$conn->query("UPDATE users SET name='$name', email='$email' WHERE id=$id");
echo "Користувача оновлено.";
?>
```

```
document.getElementById('registerForm').addEventListener('submit', async function(e)
{
```

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		4

```

e.preventDefault();

const name = document.getElementById('name').value;
const email = document.getElementById('email').value;
const password = document.getElementById('password').value;

const res = await fetch('register.php', {
  method: 'POST',
  headers: { 'Content-Type': 'application/json' },
  body: JSON.stringify({ name, email, password })
});

const data = await res.text();
alert(data);
getUsers();
});

async function getUsers() {
  const res = await fetch('get_users.php');
  const users = await res.json();
  const list = document.getElementById('userList');
  list.innerHTML = "";
  users.forEach(user => {
    const li = document.createElement('li');
    li.textContent = `${user.name} (${user.email})`;
    list.appendChild(li);
  });
}

```

<!DOCTYPE html>

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		5

```

<html lang="uk">
<head>
  <meta charset="UTF-8">
  <title>Користувачі</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
<h2>Реєстрація користувача</h2>
<form id="registerForm">
  <input type="text" id="name" placeholder="Ім'я" required>
  <input type="email" id="email" placeholder="Email" required>
  <input type="password" id="password" placeholder="Пароль" required>
  <button type="submit">Зареєструватись</button>
</form>
<hr>
<h2>Список користувачів</h2>
<button onclick="getUsers()">Оновити список</button>
<ul id="userList"></ul>
<script src="script.js"></script>
</body>
</html>

```

Результат виконання:

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		6

Реєстрація користувача

Список користувачів

- Роман (vtk241_srs@student.ztu.edu.ua)

Завдання 2: Реалізація заміток (Notes App) за допомогою FETCH та PHP

Опис завдання: Створіть веб-додаток, який дозволяє користувачам додавати, переглядати, оновлювати та видаляти замітки. Користувачі можуть створювати замітки, вказуючи заголовок та текст. Всі замітки повинні зберігатися на сервері. Взаємодія між клієнтом та сервером повинна здійснюватися за допомогою асинхронних запитів та PHP на серверній стороні.

Ключові вимоги:

1. Додавання заміток: Користувач повинен мати можливість додавати нові замітки, вказуючи заголовок та текст, та надсилати їх на сервер за допомогою асинхронних запитів.
2. Перегляд заміток: Усі замітки, збережені на сервері, повинні відображатися на головній сторінці.
3. Оновлення заміток: Користувач повинен мати можливість оновлювати замітки, змінюючи їх заголовок або текст. Зміни повинні відображатися на сервері після натискання відповідної кнопки або події.
4. Видалення заміток: Користувач повинен мати можливість видаляти замітки за допомогою відповідної опції або кнопки. Видалення заміток повинно бути асинхронним, без перезавантаження сторінки.

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		7

5. Валідація даних на клієнті та сервері: Перед надсиленням даних на сервер, вони повинні бути перевірені на клієнті (заголовок та текст мають бути заповнені).

6. Відправка та отримання даних за допомогою fetch API: Взаємодія з сервером повинна здійснюватися за допомогою fetch API на клієнтській стороні для відправки та отримання даних.

7. Зберігання даних на сервері з використанням PHP: Замітки повинні зберігатися на сервері за допомогою PHP. Використовуйте MySQL або іншу базу даних для зберігання інформації про замітки.

8. Асинхронність та безперервність додатку: Усі операції (додавання, оновлення, видалення заміток) повинні відбуватися без перезавантаження сторінки, з використанням асинхронних запитів.

```
<?php
```

```
header("Content-Type: application/json");
```

```
$data = json_decode(file_get_contents("php://input"));
```

```
if (!$data || !isset($data->title) || !isset($data->content) || !isset($data->user_id)) {
```

```
    http_response_code(400);
```

```
    echo json_encode(["status" => "error", "message" => "Неправильні або порожні  
вхідні дані"]);
```

```
    exit;
```

```
}
```

```
$conn = new mysqli("localhost", "root", "", "lab6");
```

```
$title = $conn->real_escape_string($data->title ?? "");
```

```
$content = $conn->real_escape_string($data->content ?? "");
```

```
$user_id = (int)$data->user_id;
```

```
if (trim($title) === "" || trim($content) === "") {
```

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		8


```

        echo json_encode(["status" => "error", "message" => "Заголовок і текст замітки
обов'язкові."]);
        exit;
    }

```

```

$conn->query("INSERT INTO notes (user_id, title, content) VALUES ($user_id,
$title, '$content')");
echo json_encode(["status" => "success", "message" => "Замітка додана!"]);
?>

```

```

<?php
header("Content-Type: application/json");
$data = json_decode(file_get_contents("php://input"));

if (!$data || !isset($data->id)) {
    http_response_code(400);
    echo json_encode(["status" => "error", "message" => "ID відсутній"]);
    exit;
}

```

```

$conn = new mysqli("localhost", "root", "", "lab6");
$id = (int)$data->id;
$conn->query("DELETE FROM notes WHERE id=$id");

echo json_encode(["status" => "success", "message" => "Замітку видалено."]);
?>

```

```

<?php

```

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		9

```

$conn = new mysqli("localhost", "root", "", "lab6");
$res = $conn->query("SELECT * FROM notes");
$notes = array();
while($row = $res->fetch_assoc()) {
    $notes[] = $row;
}
echo json_encode($notes);
?>

<?php
header("Content-Type: application/json");
$data = json_decode(file_get_contents("php://input"));

if (!$data || !isset($data->id) || !isset($data->title) || !isset($data->content)) {
    http_response_code(400);
    echo json_encode(["status" => "error", "message" => "Неправильні або порожні дані"]);
    exit;
}

$conn = new mysqli("localhost", "root", "", "lab6");
$id = (int)$data->id;
$title = $conn->real_escape_string($data->title ?? "");
$content = $conn->real_escape_string($data->content ?? "");

if (trim($title) === "" || trim($content) === "") {
    echo json_encode(["status" => "error", "message" => "Поля не можуть бути порожніми"]);
    exit;
}

```

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		10

```
$conn->query("UPDATE notes SET title='$title', content='$content' WHERE id=$id");
echo json_encode(["status" => "success", "message" => "Замітку оновлено."]);
?>
```

```
textarea {
    display: block;
    width: 300px;
    height: 100px;
    margin: 5px 0;
}
#notesContainer div {
    border: 1px solid #ccc;
    margin: 10px 0;
    padding: 10px;
}
```

```
document.getElementById('noteForm').addEventListener('submit', async function(e) {
    e.preventDefault();
    const title = document.getElementById('title').value;
    const content = document.getElementById('content').value;

    const res = await fetch('add_note.php', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },
        body: JSON.stringify({ title, content, user_id: 1 })
    });
```

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				11
Змн.	Арк.	№ докум.	Підпис	Дата		

```

const msg = await res.text();
alert(msg);
loadNotes();
});

async function loadNotes() {
    const res = await fetch('get_notes.php');
    const notes = await res.json();
    const container = document.getElementById('notesContainer');
    container.innerHTML = "";
    notes.forEach(note => {
        const div = document.createElement('div');
        div.innerHTML = `
            <h3>${note.title}</h3>
            <p>${note.content}</p>
            <button onclick="deleteNote(${note.id})">Видалити</button>
            <button onclick="editNote(${note.id}, '${note.title}', '${note.content}')">Реда-
гувати</button>
            <hr>
        `;
        container.appendChild(div);
    });
}

async function deleteNote(id) {
    await fetch('delete_note.php', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },
        body: JSON.stringify({ id })
    });
}

```

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		12

```

loadNotes();
}

async function editNote(id, oldTitle, oldContent) {
    const newTitle = prompt('Новий заголовок', oldTitle);
    const newContent = prompt('Новий текст', oldContent);
    if (newTitle && newContent) {
        await fetch('update_note.php', {
            method: 'POST',
            headers: { 'Content-Type': 'application/json' },
            body: JSON.stringify({ id, title: newTitle, content: newContent })
        });
        loadNotes();
    }
}

```

loadNotes();

```

<!DOCTYPE html>
<html lang="uk">
<head>
    <meta charset="UTF-8">
    <title>Замітки</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
<h2>Додати замітку</h2>
<form id="noteForm">
    <input type="text" id="title" placeholder="Заголовок" required>

```

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		13

```

<textarea id="content" placeholder="Текст замітки" required></textarea>
<button type="submit">Зберегти</button>
</form>
<hr>
<h2>Список заміток</h2>
<div id="notesContainer"></div>
<script src="script.js"></script>
</body>
</html>

```

Результат виконання:

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		14

Додати замітку

3

Список заміток

1

1

2

2

3

3

Висновок

Під час лабораторної роботи №6 я навчився працювати з асинхронними запитами на сервер за допомогою JavaScript і технології AJAX. Було реалізовано обмін даними між клієнтом і сервером без повного оновлення веб-сторінки, що значно покращує швидкість та зручність взаємодії користувача з сайтом. Також було опрацьовано отримання та обробку відповідей у форматі JSON. Отримані знання та

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				15
Змн.	Арк.	№ докум.	Підпис	Дата		

практичні навички дозволяють створювати сучасні інтерактивні інтерфейси та оптимізовані веб-додатки.

		Степанець Р. С.			Житомирська політехніка.24.121.09.000 – Лр6	Арк.
		Ковтун В.В				
Змн.	Арк.	№ докум.	Підпис	Дата		16