

## Лабораторна робота №6

### Тема: Асинхронні запити на сервер та їх обробка

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

#### 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.
- 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.

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6						
Змн.	Арк.	№ докум.	Підпис	Дата							
Розроб.		Гречко О. О.			Звіт з лабораторної роботи №6			Літ.	Арк.	Аркуші	
Перевір.		Ковтун В. В.								1	14
Реценз.								ФІКТ, гр. ІПЗ-23-3			
Н. Контр.											
Зав.каф.											

## 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.

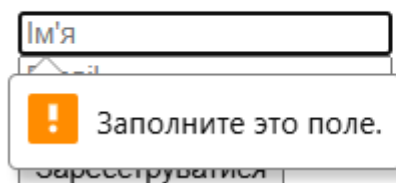
### Ключові вимоги:

- Додавання заміток:** Користувач повинен мати можливість додавати нові замітки, вказуючи заголовок та текст, та надсилати їх на сервер за допомогою асинхронних запитів.
- Перегляд заміток:** Усі замітки, збережені на сервері, повинні відображатися на головній сторінці.
- Оновлення заміток:** Користувач повинен мати можливість оновлювати замітки, змінюючи їх заголовок або текст. Зміни повинні відображатися на сервері після натискання відповідної кнопки або події.
- Видалення заміток:** Користувач повинен мати можливість видаляти замітки за допомогою відповідної опції або кнопки. Видалення заміток повинно бути асинхронним, без перезавантаження сторінки.
- Валідація даних на клієнті та сервері:** Перед надсиланням даних на сервер, вони повинні бути перевірені на клієнті (заголовок та текст мають бути заповнені).
- 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.
- 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.
- Асинхронність та безперервність додатку:** Усі операції (додавання, оновлення, видалення заміток) повинні відбуватися без перезавантаження сторінки, з використанням асинхронних запитів.

Хід роботи:

### Завдання 1:

## Реєстрація користувача



The image shows a web form for user registration. It has a text input field labeled 'Ім'я' (Name). Below the field, there is a red error message icon (an exclamation mark inside a square) and the text 'Заполните это поле.' (Fill in this field.). At the bottom of the form, there is a button labeled 'Зареєструватися' (Register).

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		2

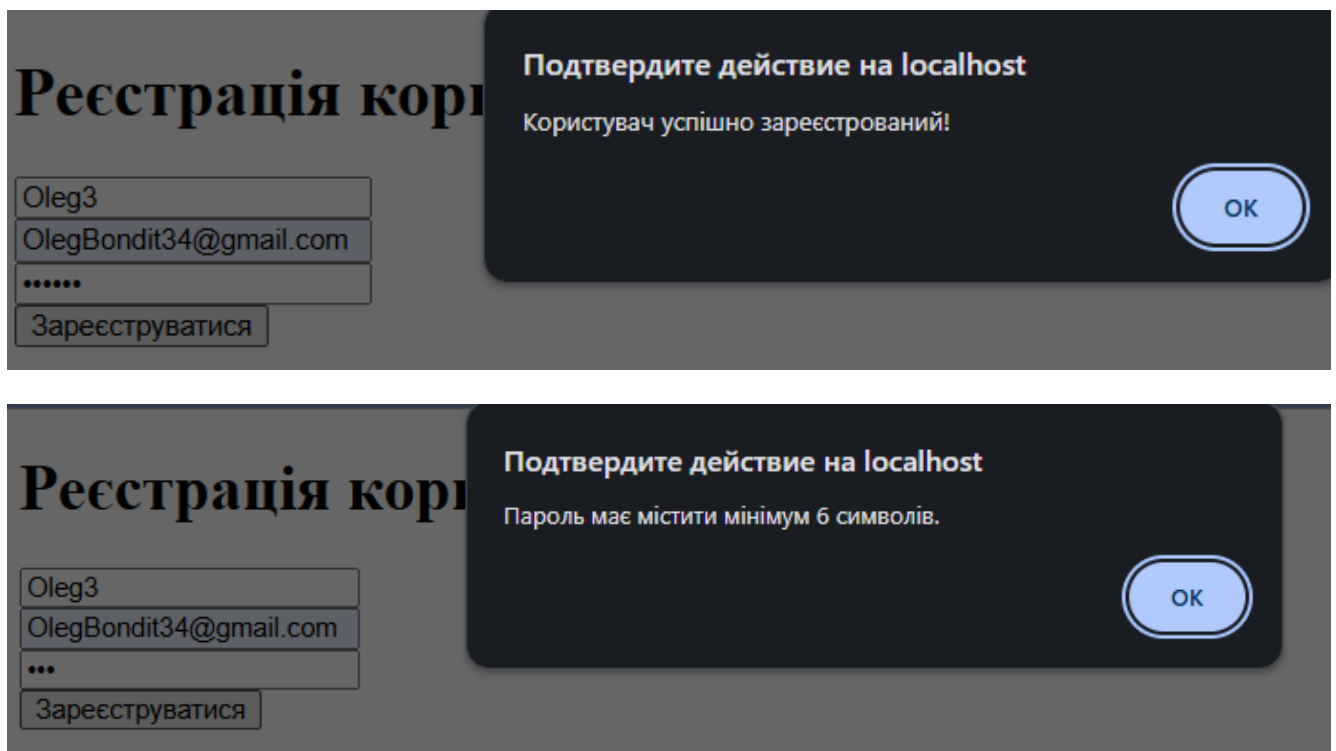


Рис. 1, 2 та 3. Завдання 1-2

## Список користувачів

Завантажити користувачів

- Oleg (Kemiral222@gmail.com) Видалити Редагувати
- Oleg2 (ipz233\_goo@student.ztu.edu.ua) Видалити Редагувати
- Oleg3 (OlegBondit34@gmail.com) Видалити Редагувати

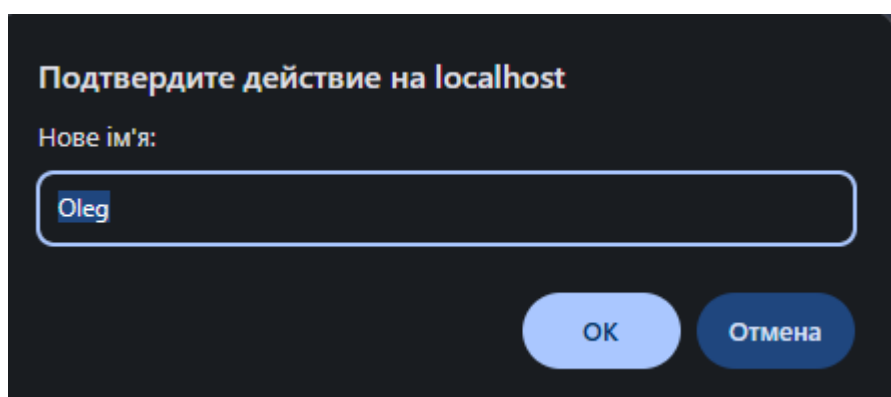
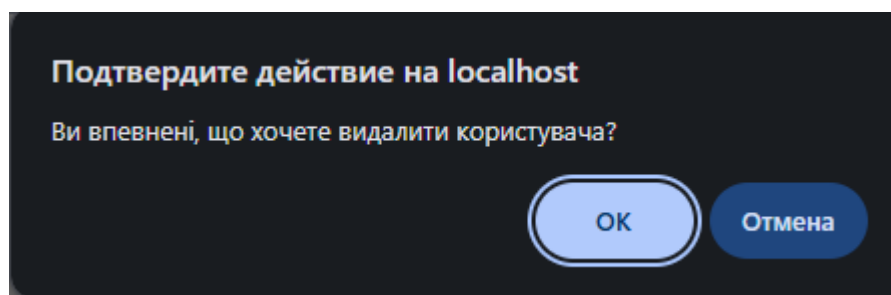


Рис. 4, 5 та 6. Завдання 3-4

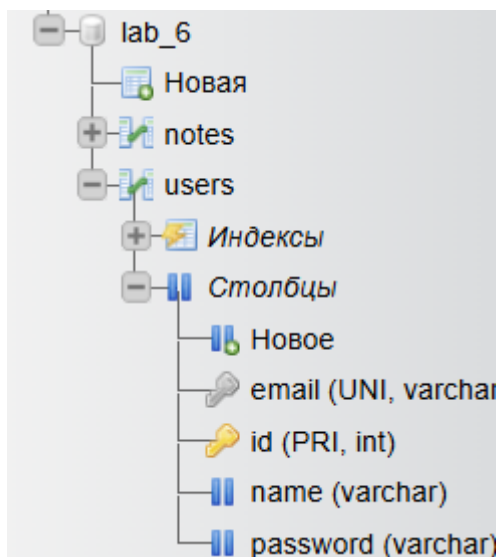


Рис. 7. БД users

### Завдання 2:

## Вхід користувача

Увійти

## Ваші нотатки

Заголовок

Зміст нотатки

Додати нотатку

- 1  
1st acc  

Редагувати

Видалити

Рис. 8. Результат виконання роботи

Після успішного логіну у нас з'явиться можливість додавати, видаляти та редагувати нотатки. Всі нотатки зберігаються в БД notes

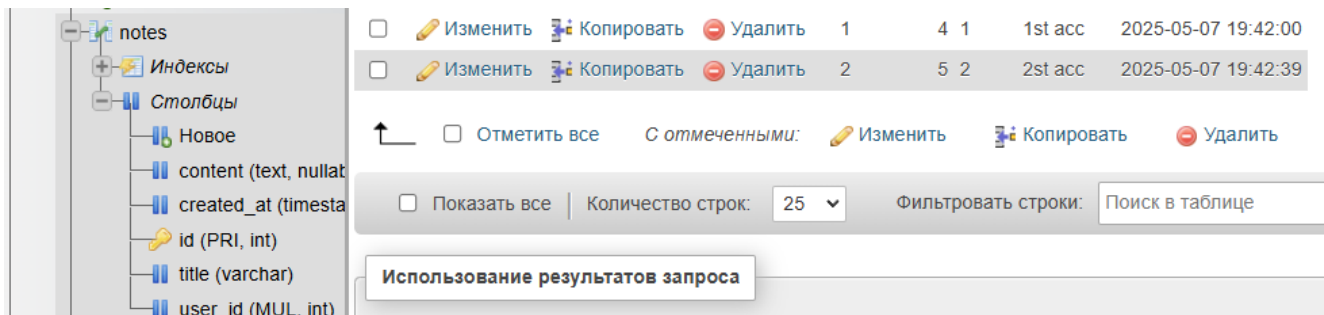


Рис. 9. БД notes

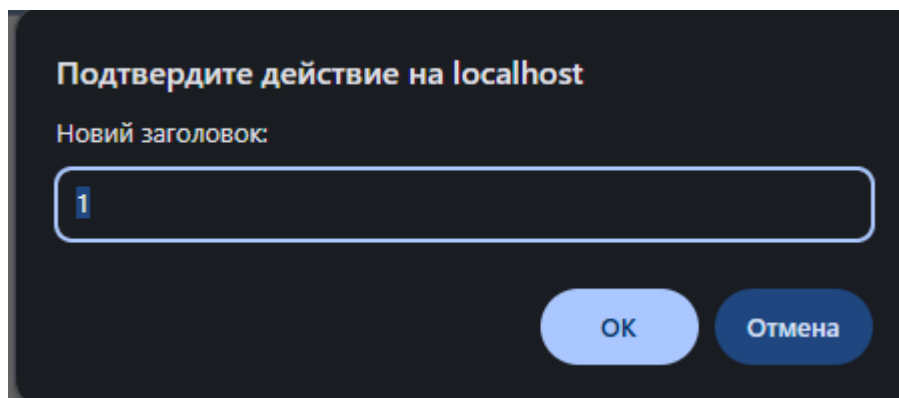


Рис. 10. Редагування запису

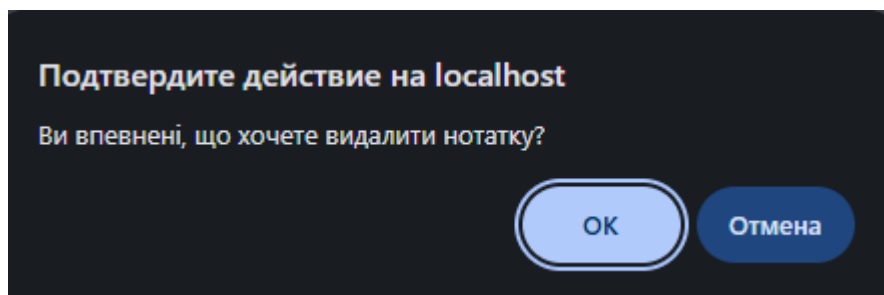


Рис. 11. Видалення запису

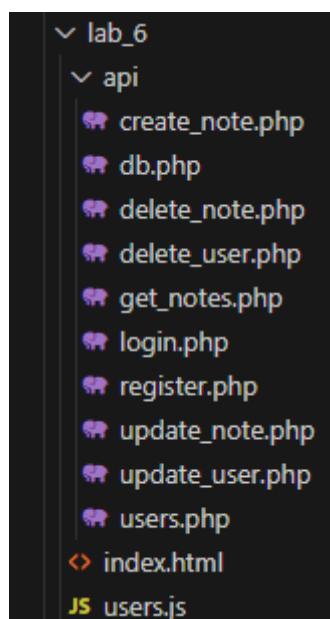


Рис. 12. Структура проекту

Лістинг (index.html):

```
<!DOCTYPE html>
<html lang="uk">
<head>
  <meta charset="UTF-8">
  <title>Користувачі</title>
</head>
<body>
  <h1>Реєстрація користувача</h1>
  <form id="registerForm">
    <input type="text" id="name" placeholder="Ім'я" required><br>
    <input type="email" id="email" placeholder="Email" required><br>
    <input type="password" id="password" placeholder="Пароль" required><br>
    <button type="submit">Зареєструватися</button>
  </form>

  <h2>Список користувачів</h2>
  <button onclick="loadUsers()">Завантажити користувачів</button>
  <ul id="userList"></ul>

  <h1>Вхід користувача</h1>
  <form id="loginForm">
    <input type="email" id="loginEmail" placeholder="Email" required><br>
    <input type="password" id="loginPassword" placeholder="Пароль" required><br>
    <button type="submit">Увійти</button>
  </form>

  <div id="notesSection" style="display:none;">
    <h2>Ваші нотатки</h2>

    <form id="noteForm">
      <input type="text" id="noteTitle" placeholder="Заголовок" required><br>
      <textarea id="noteContent" placeholder="Зміст нотатки" required></textarea><br>
      <button type="submit">Додати нотатку</button>
    </form>

    <ul id="noteList"></ul>
  </div>

  <script src="users.js"></script>
</body>
</html>
```

Лістинг (users.js):

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

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		6

```

{
    e.preventDefault();

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

    if (!name || !email || !password) {
        alert("Будь ласка, заповніть усі поля.");
        return;
    }
    //POST-запит на сервер з джсон-даними
    const response = await fetch('api/register.php', {
        method: 'POST',
        headers: {'Content-Type': 'application/json'},
        body: JSON.stringify({name, email, password})
    });

    const result = await response.json();
    alert(result.message);
});

async function loadUsers() {
    try {
        const res = await fetch('api/users.php'); //отримуєм список користувачів з
        users.php
        const users = await res.json();

        const list = document.getElementById('userList');
        list.innerHTML = '';
        users.forEach(user => {
            const li = document.createElement('li');
            li.innerHTML = `${user.name} (${user.email})
            <button onclick="deleteUser(${user.id})">Видалити</button>
            <button onclick="editUser(${user.id}, '${user.name}',
            '${user.email}')">Редагувати</button>`;

            list.appendChild(li);
        });
    } catch (err) {
        console.error("Помилка при завантаженні користувачів:", err);
    }
}

function editUser(id, currentName, currentEmail) {
    const name = prompt("Нове ім'я:", currentName);
    const email = prompt("Новий email:", currentEmail);

    if (!name || !email) return;

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7



```

    fetch('api/update_user.php', {
        method: 'POST',
        headers: {'Content-Type': 'application/json'},
        body: JSON.stringify({id, name, email})
    })
    .then(res => res.json())
    .then(result => {
        alert(result.message);
        loadUsers();
    });
}

async function deleteUser(id) {
    if (!confirm("Ви впевнені, що хочете видалити користувача?")) return;

    const res = await fetch('api/delete_user.php', {
        method: 'POST',
        headers: {'Content-Type': 'application/json'},
        body: JSON.stringify({id})
    });

    const result = await res.json();
    alert(result.message);
    loadUsers();
}

document.getElementById("loginForm").addEventListener("submit", async function (e) {
    e.preventDefault();

    const email = document.getElementById("loginEmail").value.trim();
    const password = document.getElementById("loginPassword").value.trim();

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

    const result = await res.json();
    alert(result.message);

    if (result.user) {
        localStorage.setItem('user', JSON.stringify(result.user));
        loadNotes();
    }
});
//2 завдання
document.getElementById("noteForm").addEventListener("submit", async function(e) {
    e.preventDefault();

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		



```

const user = JSON.parse(localStorage.getItem("user"));
const title = document.getElementById("noteTitle").value.trim();
const content = document.getElementById("noteContent").value.trim();

if (!title || !content) {
    alert("Заповніть усі поля.");
    return;
}

const res = await fetch("api/create_note.php", {
    method: "POST",
    headers: {'Content-Type': 'application/json'},
    body: JSON.stringify({user_id: user.id, title, content})
});

const result = await res.json();
alert(result.message);
document.getElementById("noteForm").reset();
loadNotes();
});

async function loadNotes() {
    const user = JSON.parse(localStorage.getItem("user"));
    document.getElementById("notesSection").style.display = "block";

    const res = await fetch(`api/get_notes.php?user_id=${user.id}`);
    const notes = await res.json();

    const list = document.getElementById("noteList");
    list.innerHTML = '';
    notes.forEach(note => {
        const li = document.createElement("li");
        li.innerHTML = `<strong>${note.title}</strong><br>${note.content}<br>
        <button onclick="editNote(${note.id}, '${note.title}',
        \`${note.content}\`)">Редагувати</button>
        <button onclick="deleteNote(${note.id})">Видалити</button><hr>`;
        list.appendChild(li);
    });
}

function editNote(id, currentTitle, currentContent) {
    const title = prompt("Новий заголовок:", currentTitle);
    const content = prompt("Новий вміст:", currentContent);

    if (!title || !content) return;

    fetch('api/update_note.php', {
        method: 'POST',
        headers: {'Content-Type': 'application/json'},

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        body: JSON.stringify({id, title, content})
    })
    .then(res => res.json())
    .then(result => {
        alert(result.message);
        loadNotes();
    });
}

async function deleteNote(id) {
    if (!confirm("Ви впевнені, що хочете видалити нотатку?")) return;

    const res = await fetch('api/delete_note.php', {
        method: 'POST',
        headers: {'Content-Type': 'application/json'},
        body: JSON.stringify({id})
    });

    const result = await res.json();
    alert(result.message);
    loadNotes();
}

```

Лістинг (create\_note.php):

```

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

$user_id = intval($data['user_id']);
$title = $conn->real_escape_string($data['title']);
$content = $conn->real_escape_string($data['content']);

$conn->query("INSERT INTO notes (user_id, title, content) VALUES ($user_id, '$title', '$content')");

echo json_encode(["message" => "Нотатку створено."]);
?>

```

Лістинг (db.php):

```

<?php
$host = "localhost";
$user = "root";
$password = "";
$dbname = "lab_6";

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

```
$conn = new mysqli($host, $user, $password, $dbname);

if ($conn->connect_error) {
    die("Помилка підключення: " . $conn->connect_error);
}
$conn->set_charset("utf8mb4");
?>
```

Лістинг (delete\_note.php):

```
<?php
header("Content-Type: application/json");
$data = json_decode(file_get_contents("php://input"), true);
require_once "db.php";

$id = intval($data['id']);
$conn->query("DELETE FROM notes WHERE id=$id");

echo json_encode(["message" => "Нотатку видалено."]);
?>
```

Лістинг (delete\_user.php):

```
<?php
header("Content-Type: application/json");
$data = json_decode(file_get_contents("php://input"), true);
require_once "db.php";

$id = intval($data['id']);
$conn->query("DELETE FROM users WHERE id=$id");

echo json_encode(["message" => "Користувача видалено."]);
?>
```

Лістинг (get\_notes.php):

```
<?php
header("Content-Type: application/json");
require_once "db.php";

$user_id = intval($_GET['user_id']);

$result = $conn->query("SELECT * FROM notes WHERE user_id = $user_id ORDER BY
created_at DESC");
$notes = [];
```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		11

```
while ($row = $result->fetch_assoc()) {
    $notes[] = $row;
}

echo json_encode($notes);
?>
```

#### Лістинг (login.php):

```
<?php
header("Content-Type: application/json");
$data = json_decode(file_get_contents("php://input"), true);
require_once "db.php";

$email = $conn->real_escape_string($data['email']);
$password = $data['password'];

$result = $conn->query("SELECT * FROM users WHERE email='$email'");
$user = $result->fetch_assoc();

if ($user && password_verify($password, $user['password'])) {
    echo json_encode(["message" => "Вхід успішний!", "user" => [
        "id" => $user["id"],
        "name" => $user["name"],
        "email" => $user["email"]
    ]]);
} else {
    echo json_encode(["message" => "Невірний email або пароль."]);
}
?>
```

#### Лістинг (register.php):

```
<?php
header("Content-Type: application/json");
$data = json_decode(file_get_contents("php://input"), true);
require_once "db.php";

$name = $conn->real_escape_string($data['name']);
$email = $conn->real_escape_string($data['email']);
$password = $data['password'];

if (strlen($password) < 6) {
    echo json_encode(["message" => "Пароль має містити мінімум 6 символів."]);
    exit();
}

$check = $conn->query("SELECT id FROM users WHERE email='$email'");
```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

```

if ($check->num_rows > 0) {
    echo json_encode(["message" => "Email вже зареєстрований."]);
    exit();
}

$hash = password_hash($password, PASSWORD_DEFAULT);
$conn->query("INSERT INTO users (name, email, password) VALUES ('$name', '$email', '$hash')");

echo json_encode(["message" => "Користувач успішно зареєстрований!"]);
?>

```

Лістинг (update\_note.php):

```

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

$id = intval($data['id']);
$title = $conn->real_escape_string($data['title']);
$content = $conn->real_escape_string($data['content']);

$conn->query("UPDATE notes SET title='$title', content='$content' WHERE id=$id");

echo json_encode(["message" => "Нотатку оновлено."]);
?>

```

Лістинг (update\_user.php):

```

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

$id = intval($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 json_encode(["message" => "Користувача оновлено."]);
?>

```

Лістинг (users.php):

```

<?php

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

```

header("Content-Type: application/json");
require_once "db.php";

$result = $conn->query("SELECT id, name, email FROM users");
$users = [];

while($row = $result->fetch_assoc()) {
    $users[] = $row;
}

echo json_encode($users);
?>

```

Висновок: в ході виконання роботи я ознайомився з асинхронними запитами на сервер та їх обробкою.

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.25.121.6.000 – Лр.6	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14