

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**



ІКНІ
Кафедра ПЗ

ЗВІТ

До лабораторної роботи № 5

На тему: “*Node.js як засіб написання сценаріїв*”

З дисципліни: “*Програмування в Інтернет*”

Лектор:

асистент каф. ПЗ

Степанов Д. С.

Виконав:

ст. гр. ПЗ-22

Гамела Б. А.

Перевірила:

старша викладачка каф. ПЗ

Грицай О. Д.

«__» _____ 2025 р.

Σ = _____

Тема роботи: Node.js як засіб написання сценаріїв.

Мета роботи: Засвоїти елементи програмування серверної частини застосувань мовою Node.js.

Теоретичні відомості

Node.js — це відкрите кросплатформове середовище виконання JavaScript, яке працює на сервері та базується на двигуні V8 від Google. Воно призначене для створення швидких і масштабованих веб-серверів, хоча застосовується і в інших сферах.

Основні особливості:

1. **Висока швидкість:** Код на Node.js може виконуватися швидше, ніж у багатьох компільованих (C, Java) чи інтерпретованих (Python, Ruby) мовах завдяки асинхронній моделі.
2. **Асинхронність і неблокуючий код:** Node.js використовує єдиний потік, коллбеки та подієво-орієнтований підхід, що дозволяє виконувати операції (наприклад, мережеві запити) без зупинки програми.
3. **Подієва модель:** Аналогічно до браузерів, Node.js обробляє події (наприклад, запити чи кліки) у реальному часі, що ідеально для інтерактивних застосунків.
4. **Екосистема npm:** Менеджер пакетів npm надає доступ до понад 500 000 open-source бібліотек, що спрощують розробку.

Приклад роботи: Простіший веб-сервер на Node.js створюється з використанням модуля http. Код запускається командою `node server.js` і відображає текст (наприклад, "Hello World") за адресою, вказаною в браузері.

Популярні бібліотеки:

- **Express:** Мінімалістичний фреймворк для створення веб-серверів.
- **Socket.io:** Інструмент для розробки програм реального часу, таких як чати.

Node.js є низькорівневою платформою, але завдяки бібліотекам значно спрощує розробку серверних застосунків.

Формулювання завдання

1. Розробити web-сторінку згідно [макета](#) (wireframe).
2. Сторінка повинна відповідати наступним [вимогам](#).

Результати виконання (спільні 5-6 лабораторна)

Репозиторій проекту на github.com: [PI-Labs](https://github.com/PI-Labs)

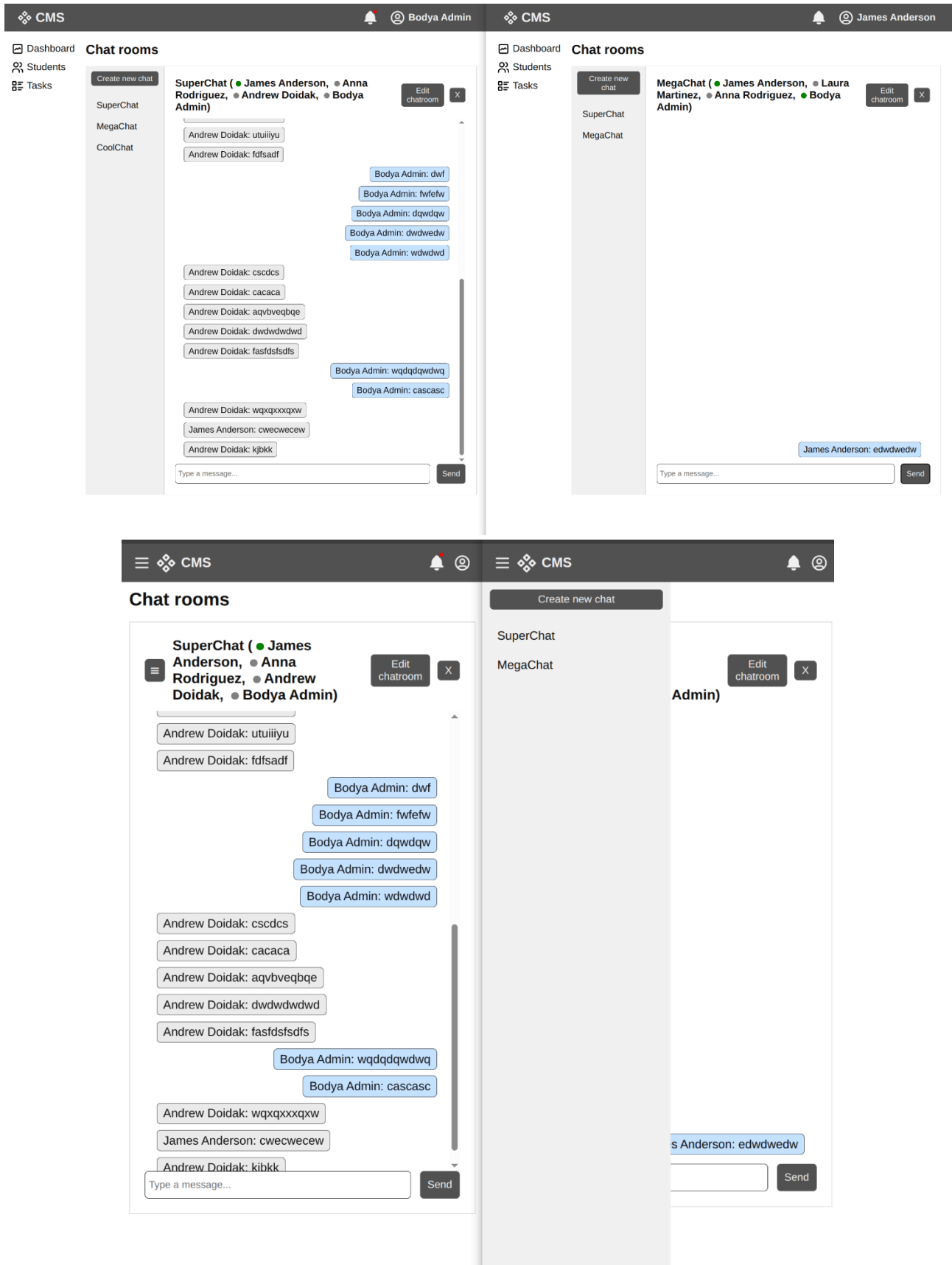


Рис. 1-2. Екран чатів

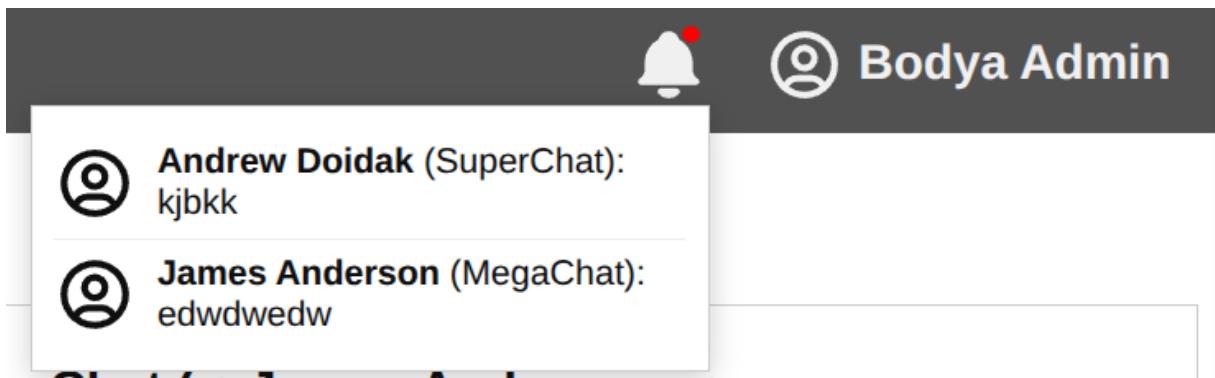


Рис. 3. Сповіщення про нові повідомлення

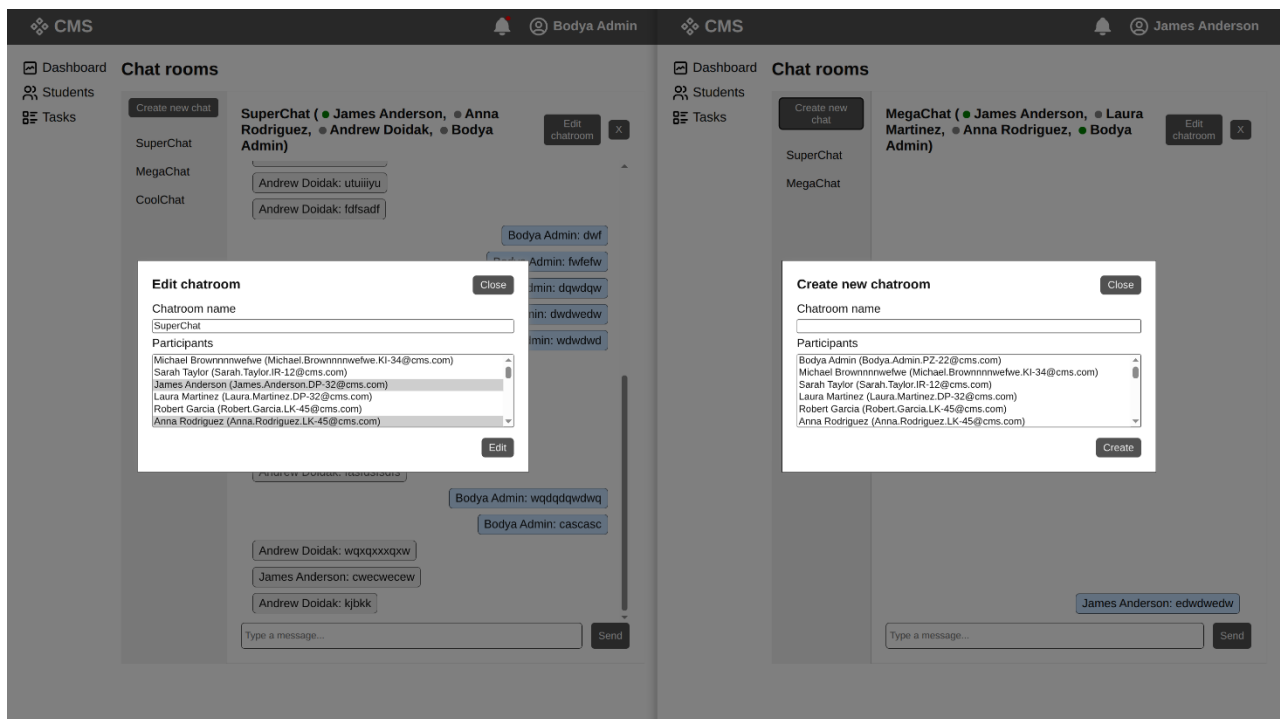


Рис. 4. Модальні вікна

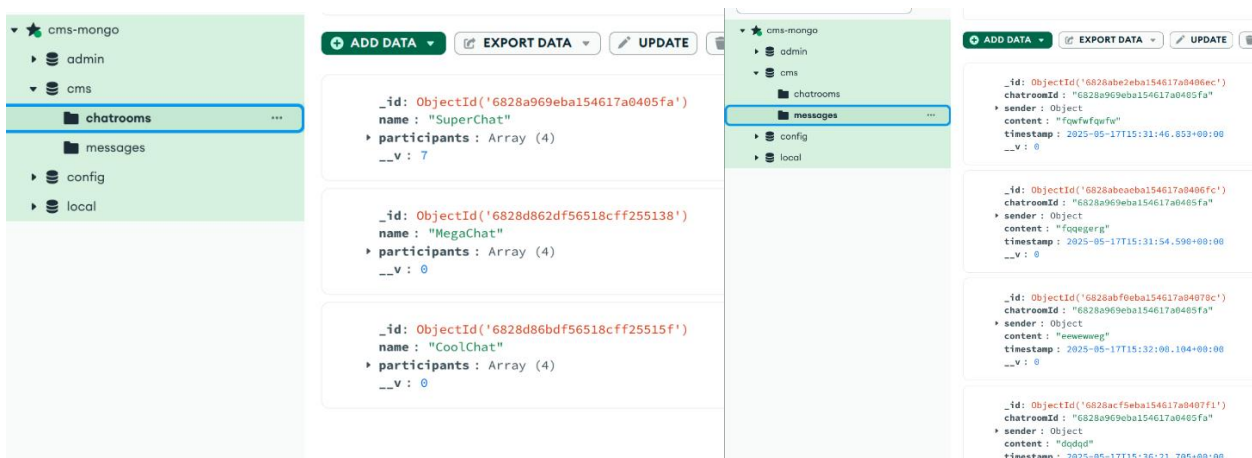


Рис. 5. База даних

Висновки (спільні 5-6 лабораторна)

Під час виконання лабораторних робіт №5 та №6 я поглибив знання з серверної веб-розробки з використанням Node.js, MongoDB та бібліотеки Socket.io для створення застосунків реального часу. Було розроблено чат-додаток, який включає клієнтську частину з формою для надсилання повідомлень та відображення чатів, а також серверну частину, що обробляє повідомлення, перевіряє їх адресатів (окремий студент, група чи всі) і зберігає дані в MongoDB. Використано Mongoose для роботи з базою даних, включаючи створення схем (повідомлення, чати), валідацію даних і зв'язки між колекціями. Для авторизації застосовано JWT, а Socket.io забезпечило миттєву доставку повідомлень і оновлення статусу учасників.

Робота розвинула навички асинхронного програмування в Node.js, роботи з NoSQL базами даних через Mongoose, реалізації реального часу комунікації та інтеграції клієнт-серверних технологій. Ці компетенції сприятимуть подальшому розвитку в створенні масштабованих веб-додатків.