Міністерство освіти і науки

Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Кафедра автоматизації проектування енергетичних процесів і систем

ЗВІТ   
про виконання лабораторної роботи №2  
з дисципліни «Програмне забезпечення мереж передачі даних»

Тема: «Анонімний веб-чат»

Виконав студент 3-го курсу

групи ТІ-92 Черноусов Денис Ігорович

Перевірила Гусєва І. І.

КИЇВ 2022

**Аналоги**

Знайти аналогічні веб застосунки достатньо легко. Найпопулярнішим на території пострадянських країн є Nekto.me. Він написаний на мові програмування JavaScript. Використовується бібліотека jQuery 1.4.1, що допомагає надсилати та отримувати повідомлення без перезавантаження сторінки. Важливо підмітити, що версія цієї бібліотеки застаріла, бо на сьогодні використовується версія 3.\*.\*.

Для того щоб розпочати переписуватися з якоюсь випадковою людиною можна ввести деяку інформацію: свою стать, свій вік, тип розмови та інше. При відправлені та отримані повідомлення, як очікувалася, сторінка не перезагружається.

Український аналог - chat.ziznannya.org. Він використовує той самий jQuery 1.12.1, бібліотеки MomentJS та RespondJS, з цього стає зрозуміло, що серверна частина сайту написана на JavaScript. Клієнтська частина написана на фрейворці AngularJS. На використали також інтерфейсний фреймворк Bootstrap це для розробки адаптивних веб-сторінок з акцентом на мобільних користувачів. Він складається з HTML, CSS і JavaScript. Bootstrap включає спеціальні компоненти CSS і плагіни jQuery.

Вище описана інформація отримана за допомогою веб розширення whatruns.com. Анонімні чати менш популярні, порівнюючи з іншими відео чатами, тож відповідно немає інформаційних статей про ці анонімні чати.

Для роботи з сокетами на клієнтській частині, що написана на JavaScript, зазвичай використовують бібліотеки WebSocket та SocketIO. Якщо серверна частина написана на іншій мові, то потрібно підбирати сокетну бібліотеку, що буде коректно працювати.

**Огляд програмного забезпечення**

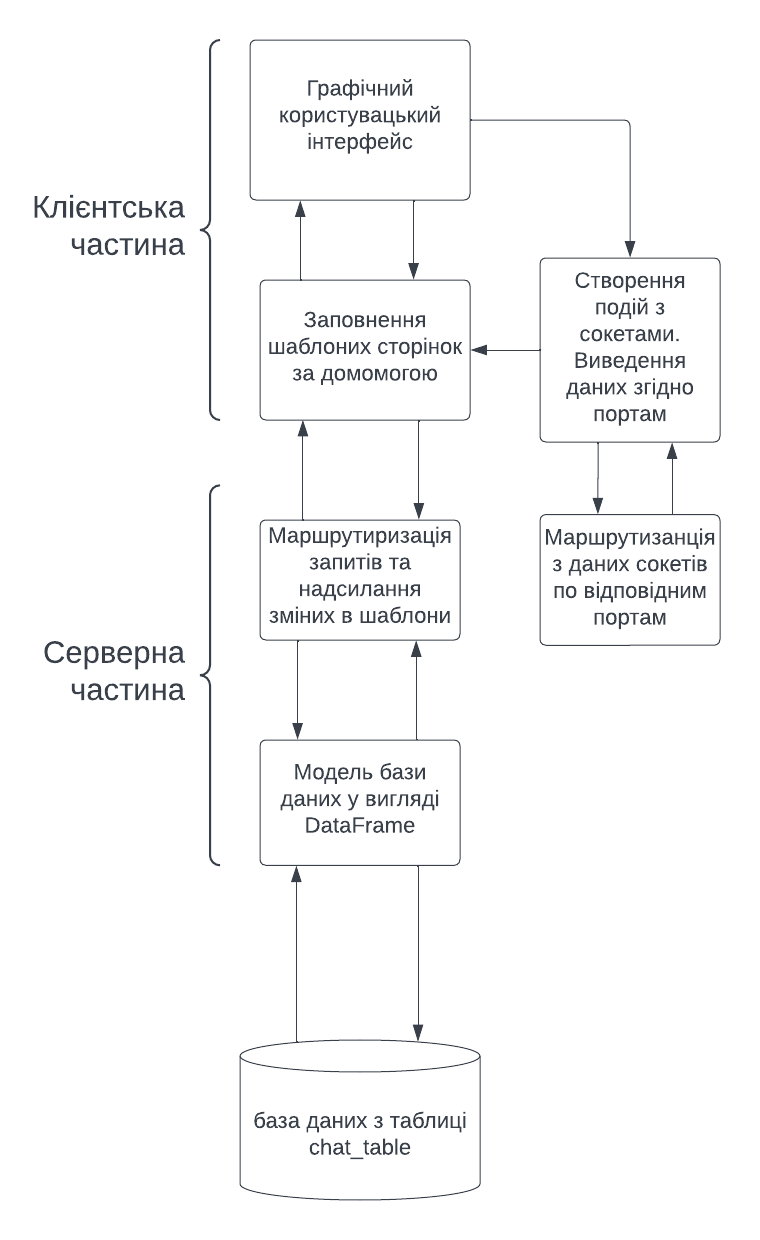
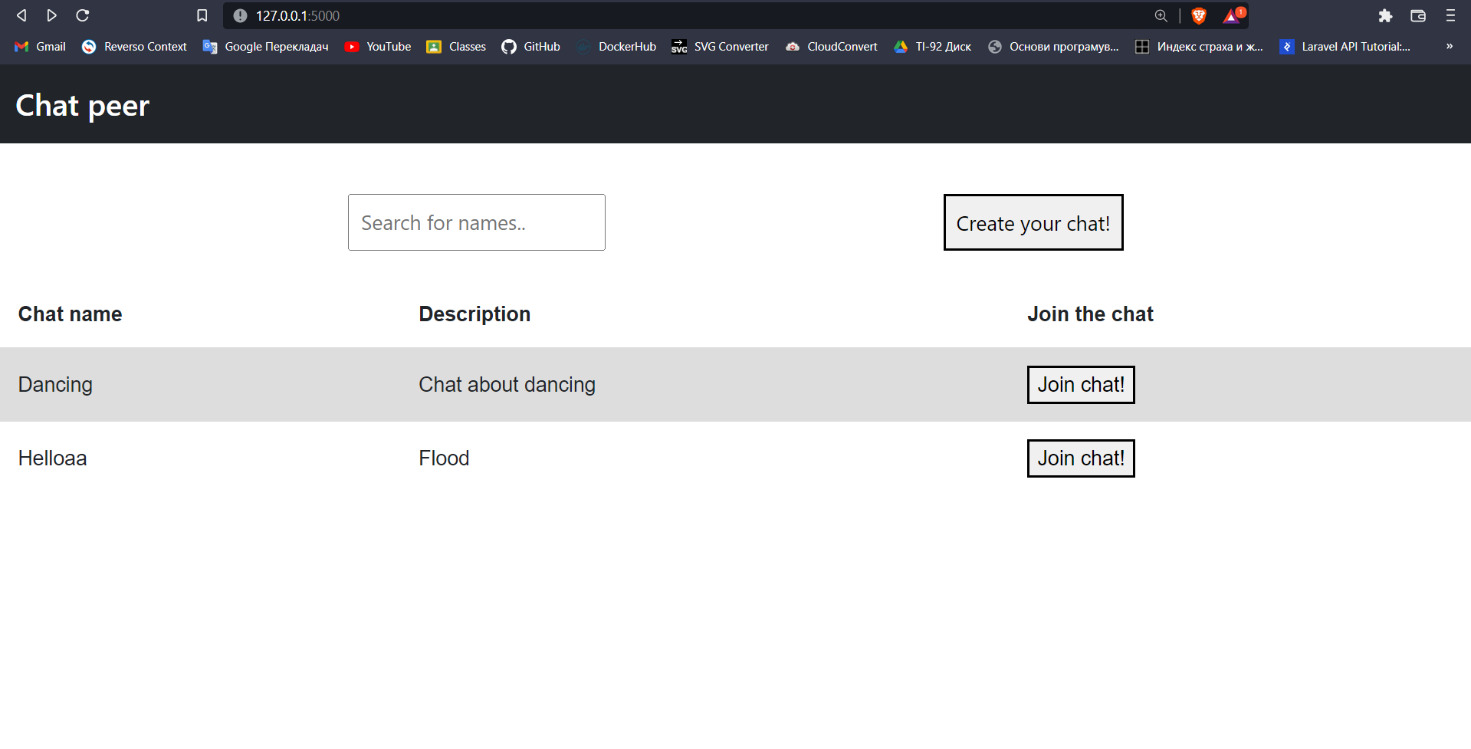
 Архітектура програмного забезпечення складається з трьох рівнів: сервера, клієнта та база даних. Діаграма зазначена на рисунку 1.1

Рисунок 1.1. Діаграма архітектури ПЗ

 Клієнтська частина доступна через браузер. На головній сторінці знаходиться таблиця кімнат для анонімної переписки. Як можна побачити з рисунку 1.2, кожна кімната містить ім’я (Chat name) та опис (Description). У відповідному рядку знаходиться кнопка за допомогою якої можна перейти в кімнату. Можна провести фільтрацію кімнат за іменами (рисунок 1.3). За допомогою кнопки “Create your chat” можна створити власну кімнату.

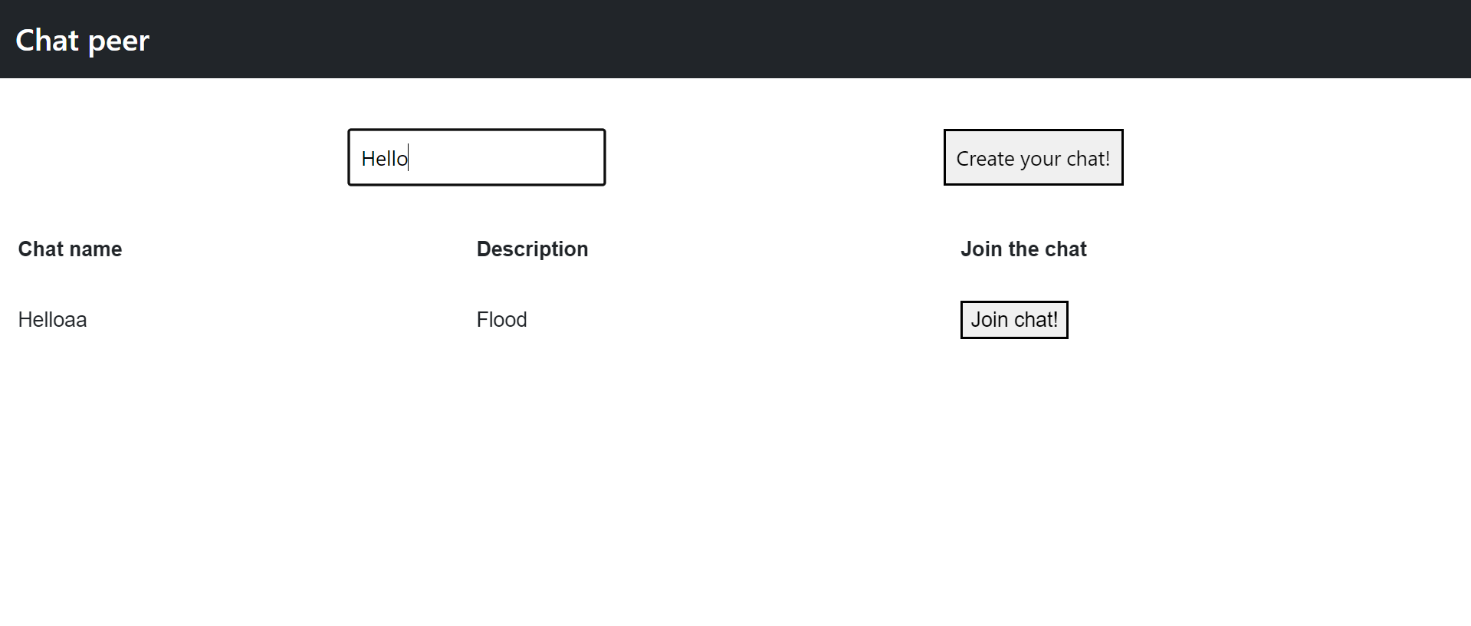
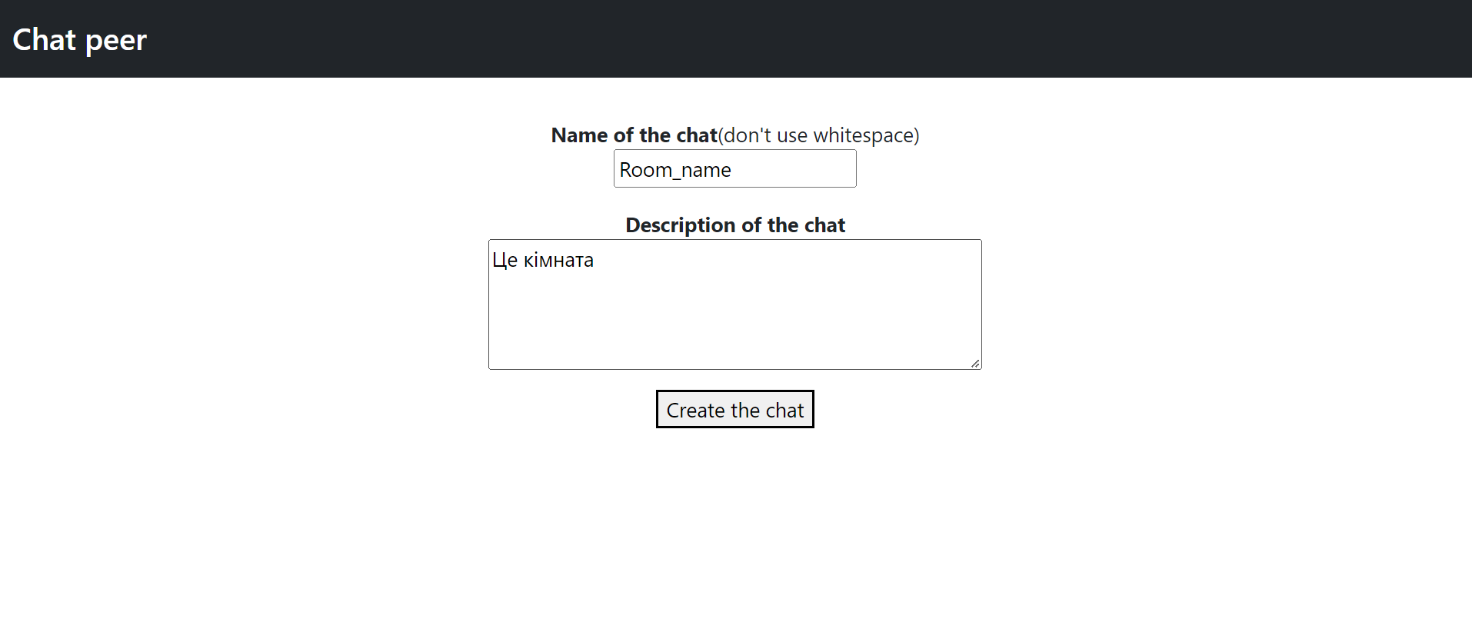
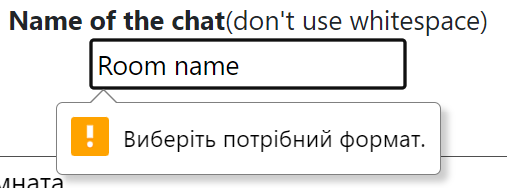
Рисунок 1.2. Головна сторінка

Рисунок 1.3. Робота фільтру кімнат

Користувач може створити свою кімнату (рисунок 1.4). Назва кімнати не повинна містити пробілів (див рисунок 1.5) та має містити від 3 до 20 символів. Опис чату - це обов’язкове поле яке має містити хоч 1 символ. Реалізована більш складніша умова – назва чату повинна бути унікальна (відбувається перевірка по базі даних) інакше надсилається повідомлення (див рисунок 1.6).

Рисунок 1.4. Сторінка створення кімнати чату

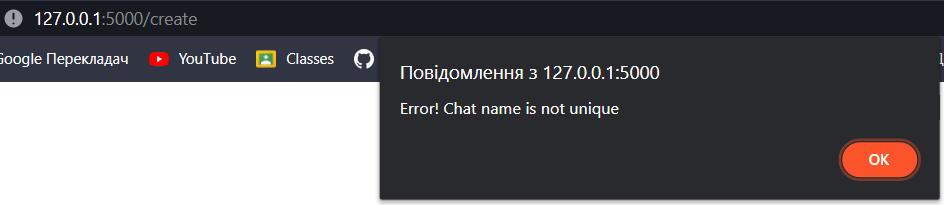
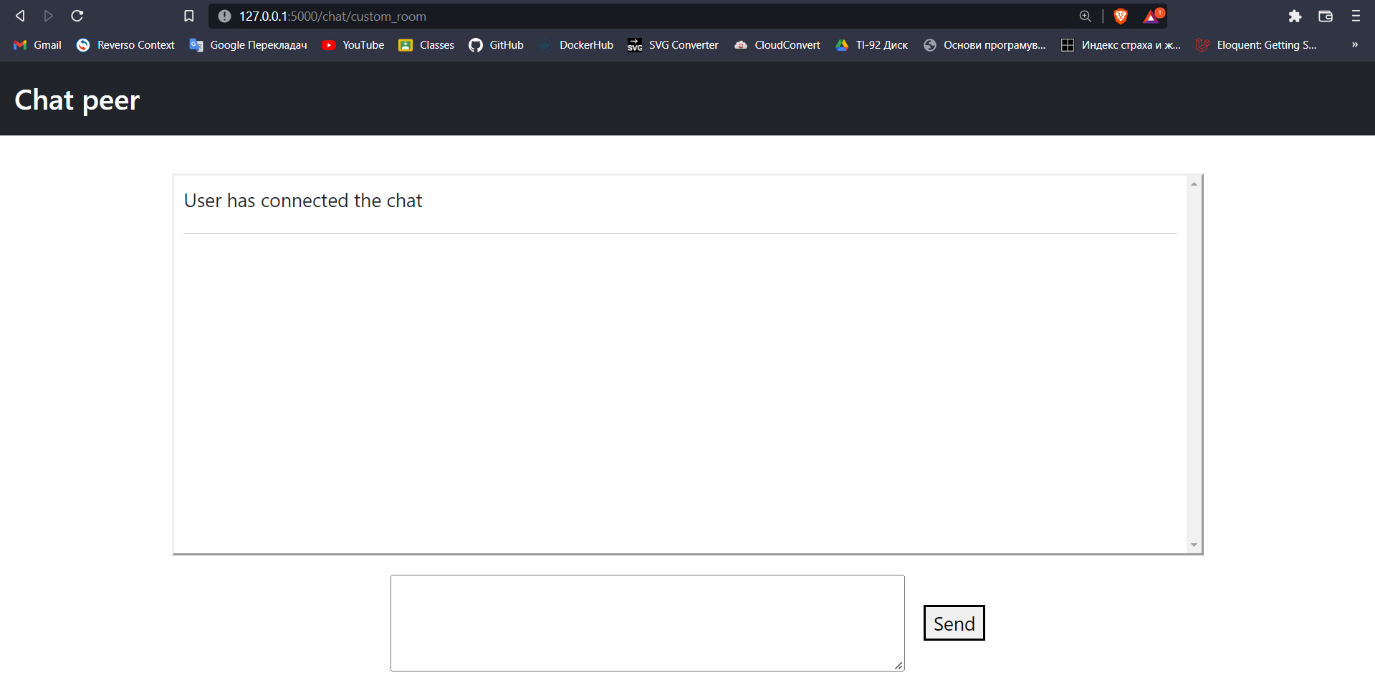
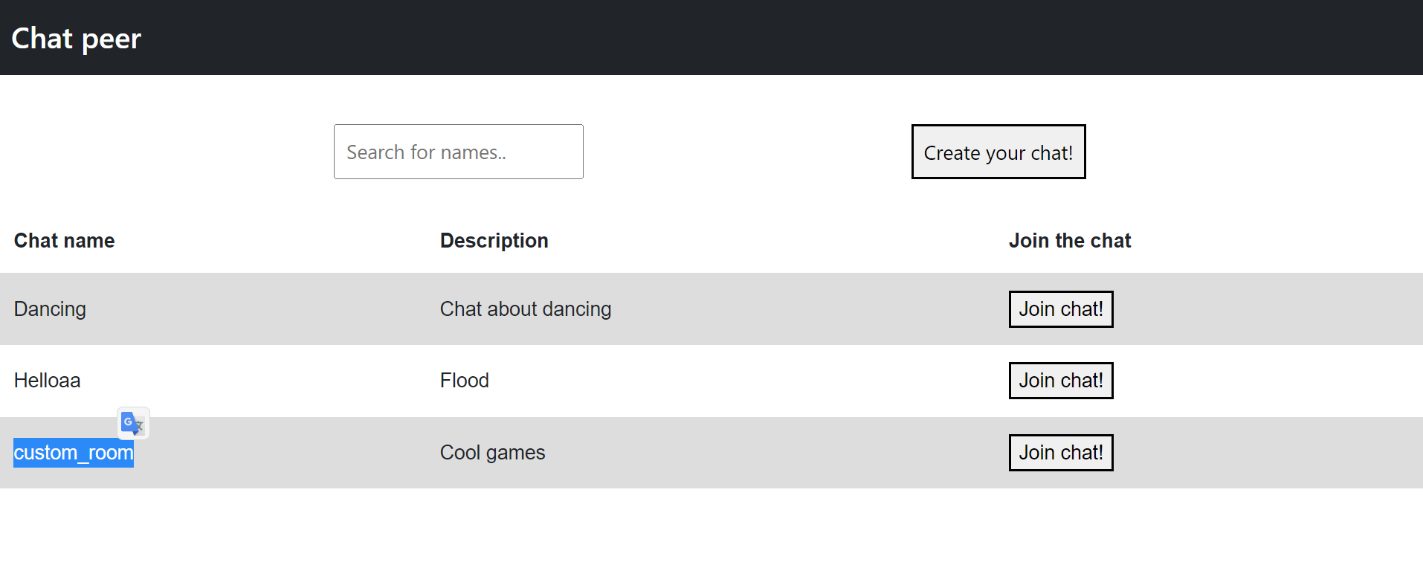
Рисунок 1.5. Назва кімнати має пройти перевірку регулярним виразом

Рисунок 1.6.

Після вдалого створення кімнати, творець кімнати перенаправляється в новостворений чат (рисунок 1.7). Відповідно ця кімната появляється в таблиці на головній сторінці (рисунок 1.8). Тепер користувач зайде в цей чат і відправить повідомлення (рисунок 1.9). Якщо користувач пише повідомлення, то перед ним буде написано “Me ”. А перед отриманим повідомленням – ні (рисунок 1.10).

Учасники в різних кімнатах не мають зв’язок між собою. Зверніть увагу на посилання та сам рисунок 1.11. В кімнаті може находитися безліч людей, що видно з рисунку (рисунок 1.12).

Рисунок 1.7. Новостворений чат

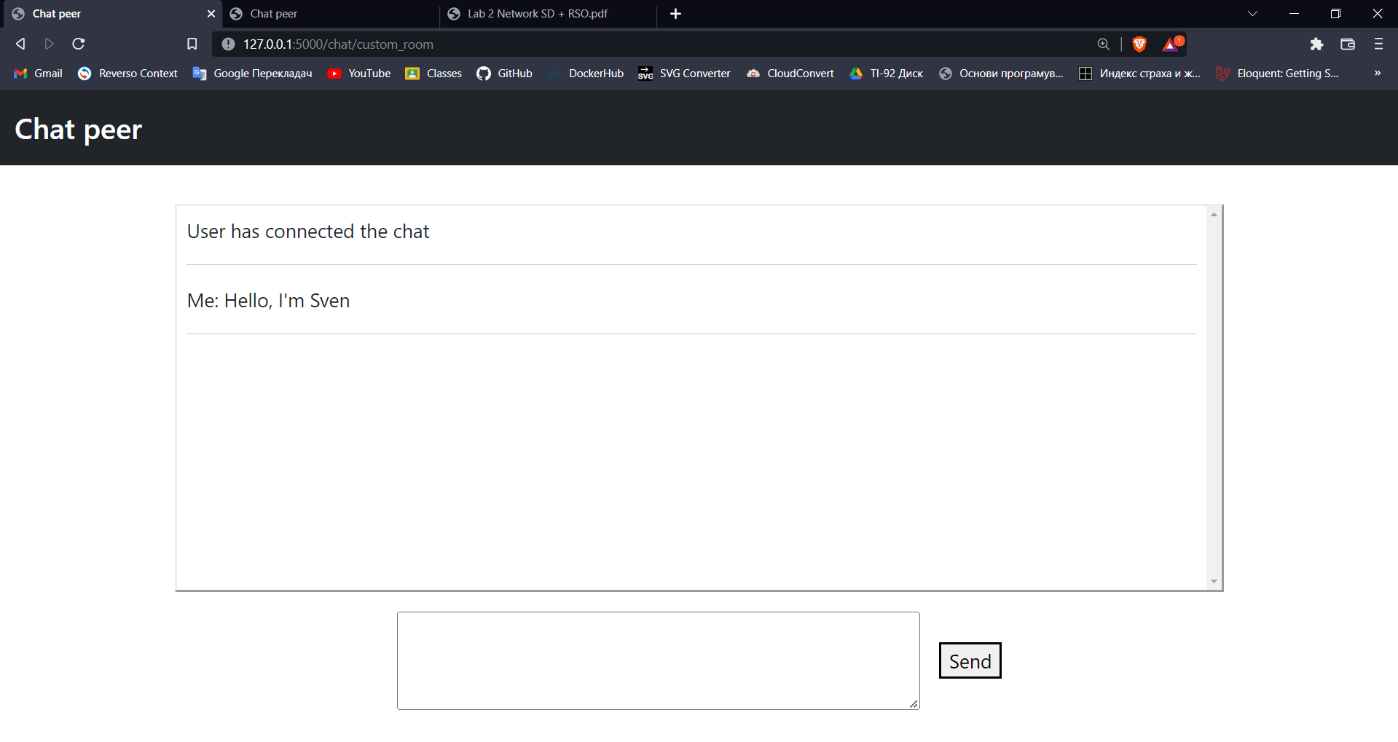
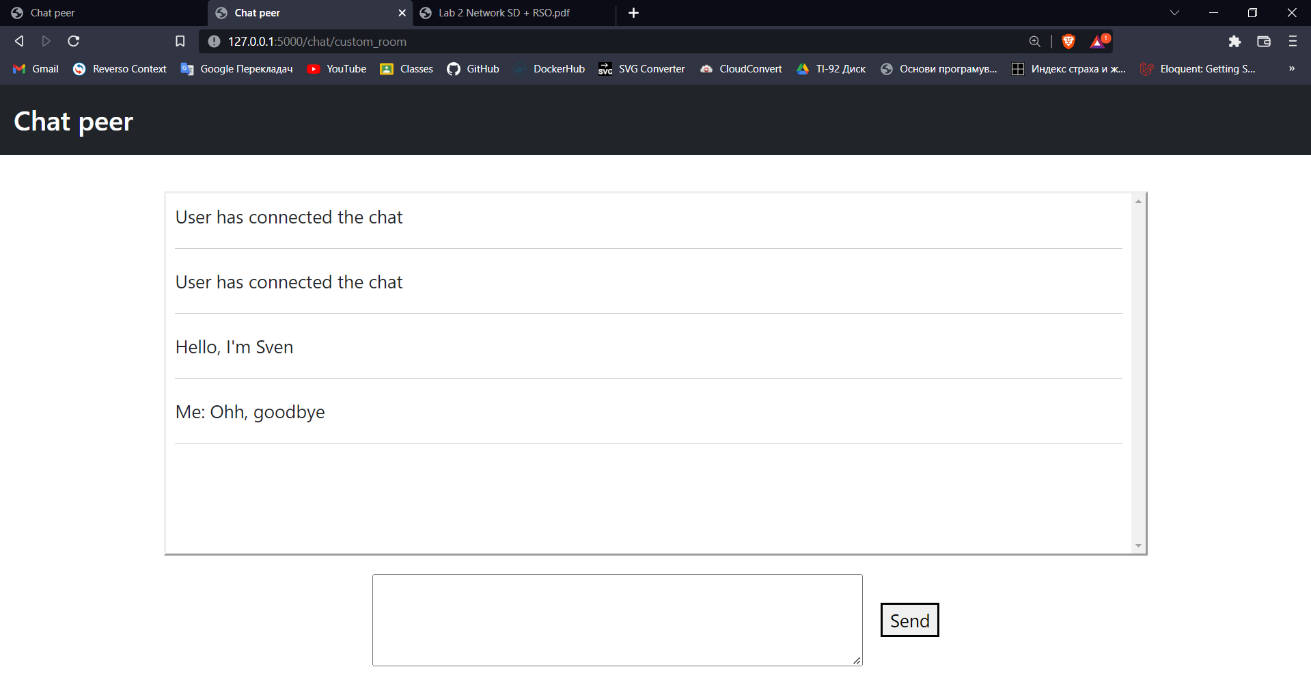
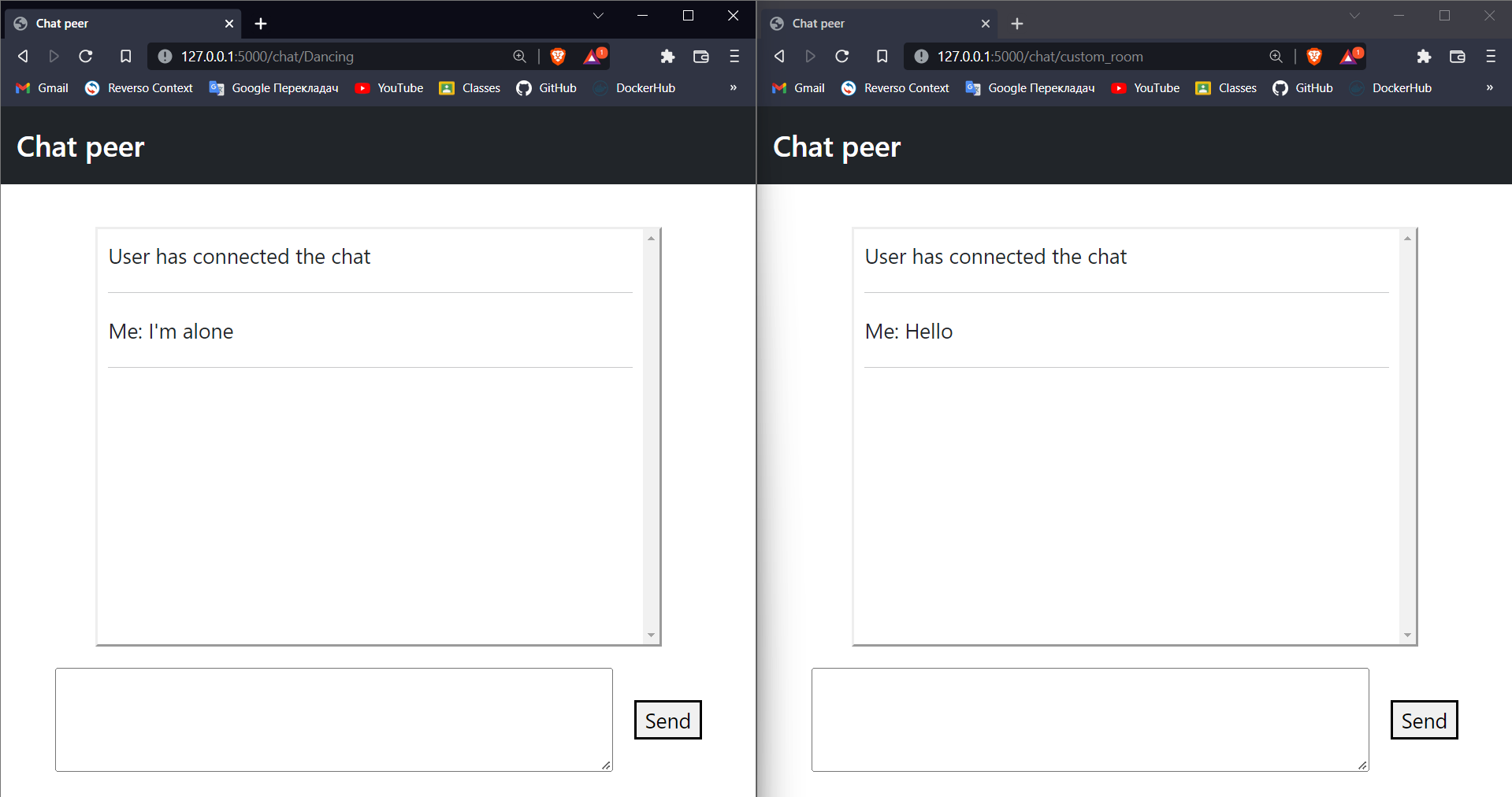
Рисунок 1.8. Добавилась кімната в таблицю

Рисунок 1.9. Написано 1ше повідомлення

Рисунок 1.10. Отримано повідомлення і написано нове

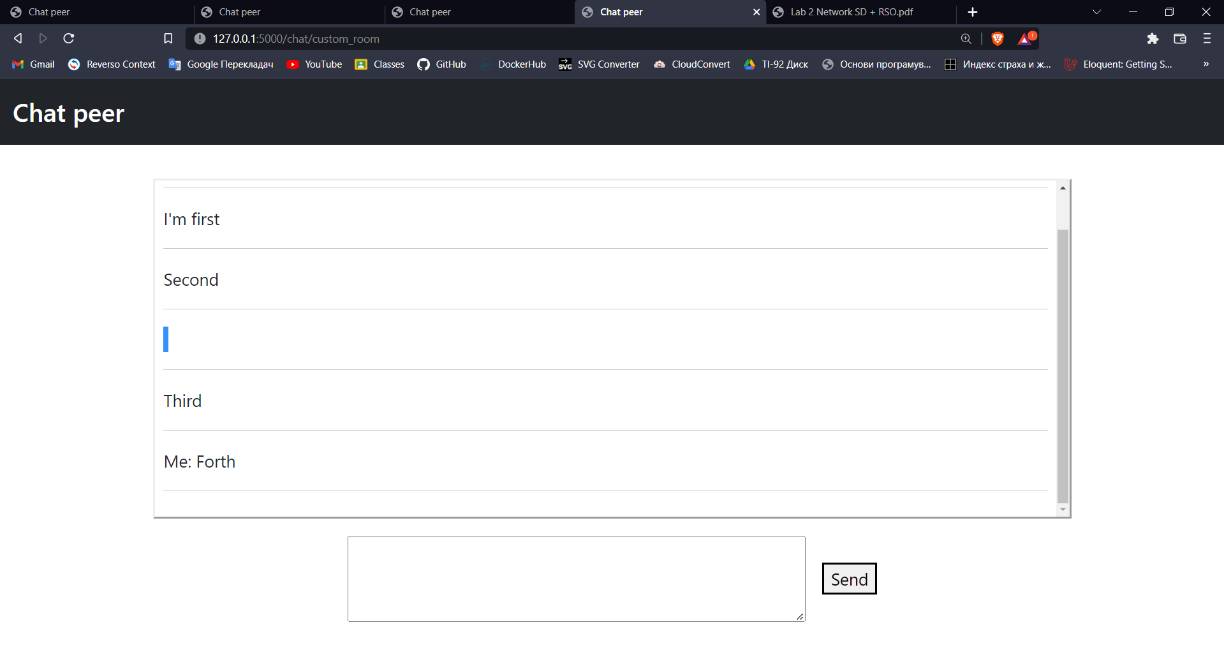
Рисунок 1.11 Учасники в різних кімнатах

Рисунок 1.12. Безліч учасників у кімнаті

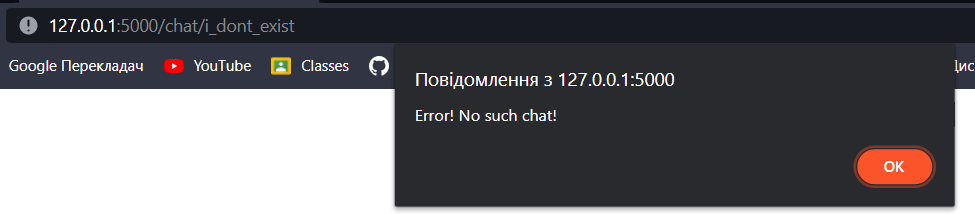
 Якщо чату не існує, то буде виведене відповідне повідомлення (див. Рисунок 1.13)

Рисунок 1.13 Немає такого чату

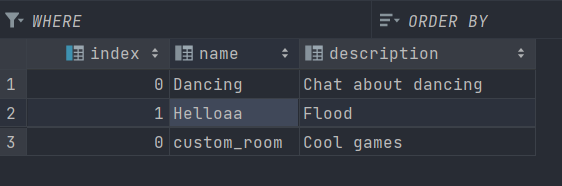
 База даних складається з однієї таблиці. В ній містяться унікальні імена анонімних кімнат та відповідно описи цих анонімних кімнат (Рисунок 1.14). Значення в стовпці name (ім’я) мають бути унікальними.

Рисунок 1.14. Таблиця чатів

**Реалізація програмного забезпечення**

Для реалізації програмного забезпечення були використані такі технології:

1. SQLite – мініатюрна реляційна база даних, що знаходиться в одному файлі exchanges.sqlite. Ця база даних не містить багатьох особливостей, наприклад можливість вказати стовпчик унікальним, проте це реалізується на серверній частині програмного забезпечення.
2. Pandas – бібліотека python для створення моделі бази даних в DataFrame
3. Flask – мікро фреймворк для розробки веб-додатків на мові програмування python.
4. Flask\_socketIO бібліотека python для роботи з сокетами на серверній стороні, що надає програмам Flask доступ до двостороннього зв'язку з низькою затримкою між клієнтами та сервером. Клієнтська програма може використовувати клієнтську бібліотеку SocketIO в Javascript
5. SocketIO – бібліотека JavaScript для роботи з сокетами на клієнтській стороні, що забезпечує двонаправлений зв'язок на основі подій у реальному часі.
6. JQuery – бібліотека JavaScript для динамічного виведення повідомлень без потреби в перезагрузки сторінки.
7. Bootstrap — це найпопулярніший фреймворк CSS для розробки веб-сайтів. Орієнтований на зовнішній вигляд клієнтської сторони

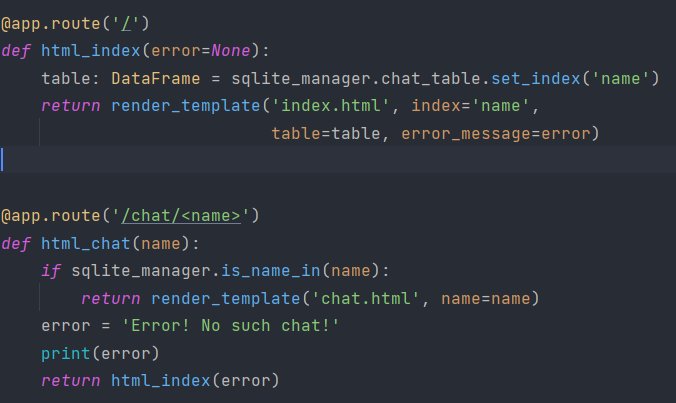
 Навігація на сайті організована подібним чином через Flask переносячи зміні в шаблони (рисунок 2.1)

Рисунок 2.1. Навігація на сайті

 Повідомлення спершу надсилаються через події разом з даними через клієнтську частину, що написана на javaScript. Серверна частина підхоплює ці події, перерозподіляє їх по кімнатам та викликає необхідні події на клієнтській частині (рисунок 2.2)

Рисунок 2.2. Сокетна архітектура

Подивитися повний код програми можна на репозиторії <https://github.com/Darkinowls/chat-peer>