Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота №9
з дисципліни **«Технології розроблення програмного забезпечення»**

на тему «різні види взаємодії додатків:
client-server, peer-to-peer, service-oriented architecture»
Тема для лабораторного циклу «HTTP-сервер»

Виконала

студентка групи ІА-04
Глушко
Юлія Петрівна

Перевірив

Колеснік Валерій
Миколайович

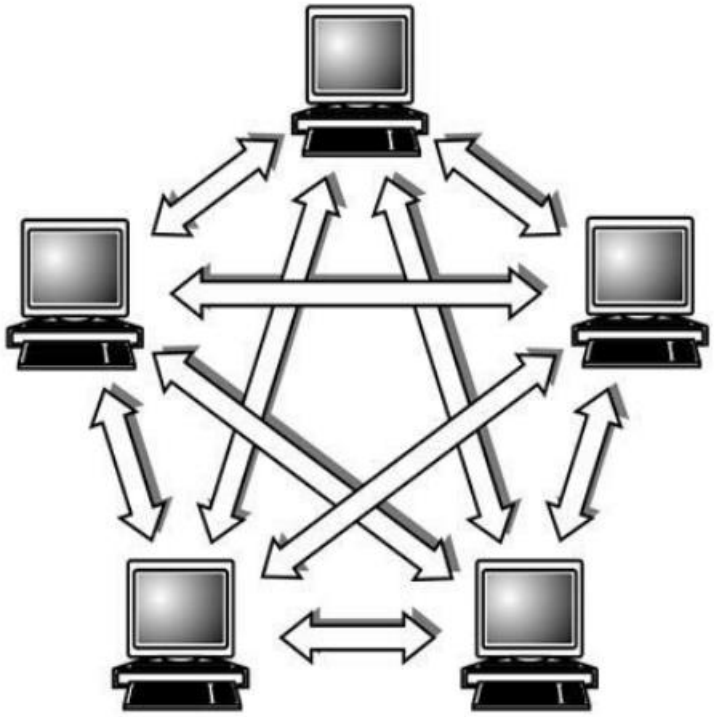
Защищено з балом _____

Київ 2022

Хід роботи :

*Додатки типу «**peer-to-peer**»*

Теоретичні відомості



Модель взаємодії додатків типу peer - to - peer має на увазі рівноправ'я клієнтських програм і відсутність серверної програми. Усі клієнтські програми контактують між собою для виконання спільних цілей. У таких мережах найчастіше виникають наступні проблеми: синхронізація даних, пошук клієнтських застосувань.

Для пошуку клієнтських застосувань може використовуватися одна загальна адреса, що містить набір адрес підключених клієнтів. Це в деякому роді нагадує сервер. Проте сервер є лише центром сходження безлічі клієнтів, а адреси можуть зберігатися в самих клієнтських застосуваннях. Це називається структуровані однорангові мережі.

Для вирішення проблеми синхронізації даних використовуються традиційні підходи: можливий варіант договору про набір для синхронізації (negotiation), або визначення по деякому алгоритму порівняння даних (hash-алгоритми і інші). Як правило, peer - to - peer застосування є звичайними застосуваннями з продуманою інфраструктурою обміну повідомленнями між клієнтами. Для таких випадків як правило розробляються спеціальні формати і протоколи обміну для структуризації діалогу між клієнтами.

Застосування при реалізації лабораторного циклу

При виконання лабораторного циклу, HTTP-сервер було реалізовано за p2p архітектурою. Лабораторна робота складається із 3 модулів: Клієнтів, Серверу (без якого все також чудово працює. Сервер потрібен лише для того, щоб peer`ам було легше знайти активні підключення) та інформації про мережу (зв'язок між клієнтами (один із одним) та клієнтами із сервером) .

Користувачів можна створити декілька, Сервер для їхнього зв'язку – лише один раз.

Кожен peer одночасно є і сервером і клієнтом. При створенні його призначається випадковий порт, який він має прослуховувати як сервер. Якщо на цей порт приходить запит, то він обробляється і генерується відповідь. Запити та відповіді по HTTP/1.1 протоколу.

Сервер працює за такою схемою :

Через графічний інтерфейс передаються потрібні дані (Рис.1), а саме , метод, який потрібно використати (get або post)(регістр не має різниці), url сторінки та порт, на який потрібно зробити запит . Дані валідуються(рис.2,3). Наступні поля – це параметри/значення, які потрібно передати . Також можна побачити номер порта даного peer та кнопку для відправки даних.

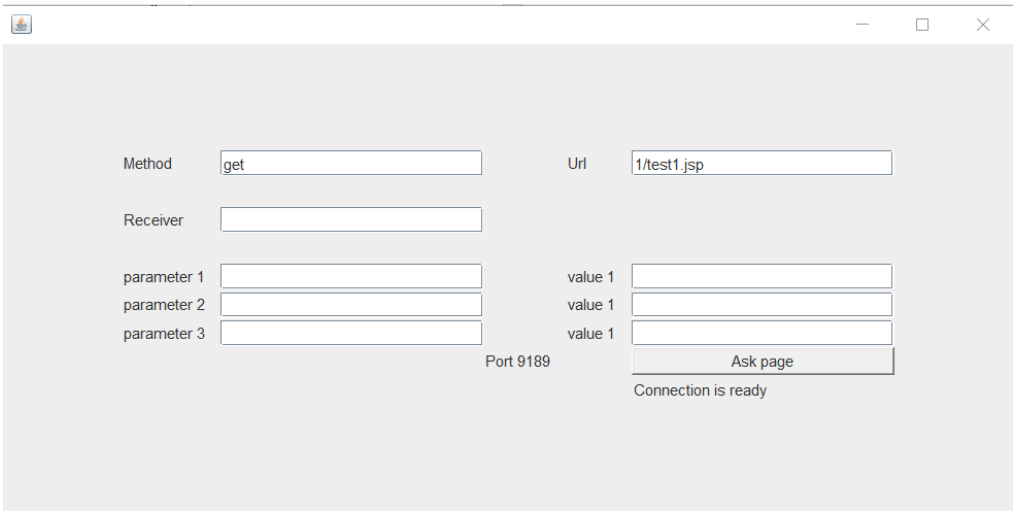


Рис.1

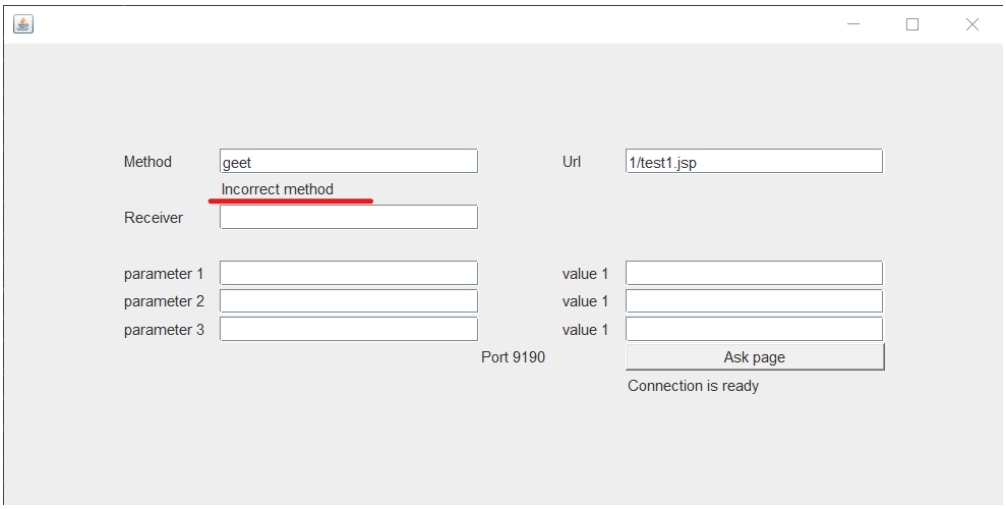


Рис.2

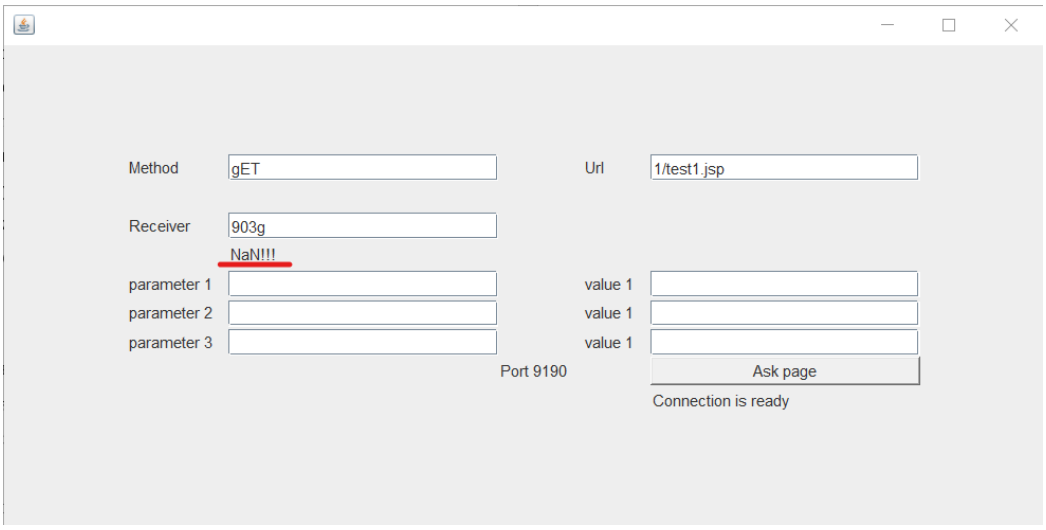


Рис.3

В подальшому створюється з’єднання клієнта з сервером (певним peer або сервером для пошуку з’єднань). Результати роботи HTTP -відповідь від сервера . Випадок із успішною процедурою (рис.4), випадок, коли сторінки не знайдено (рис.5) та випадок, коли намагаємось підключитись до не активного peer (рис. 6)

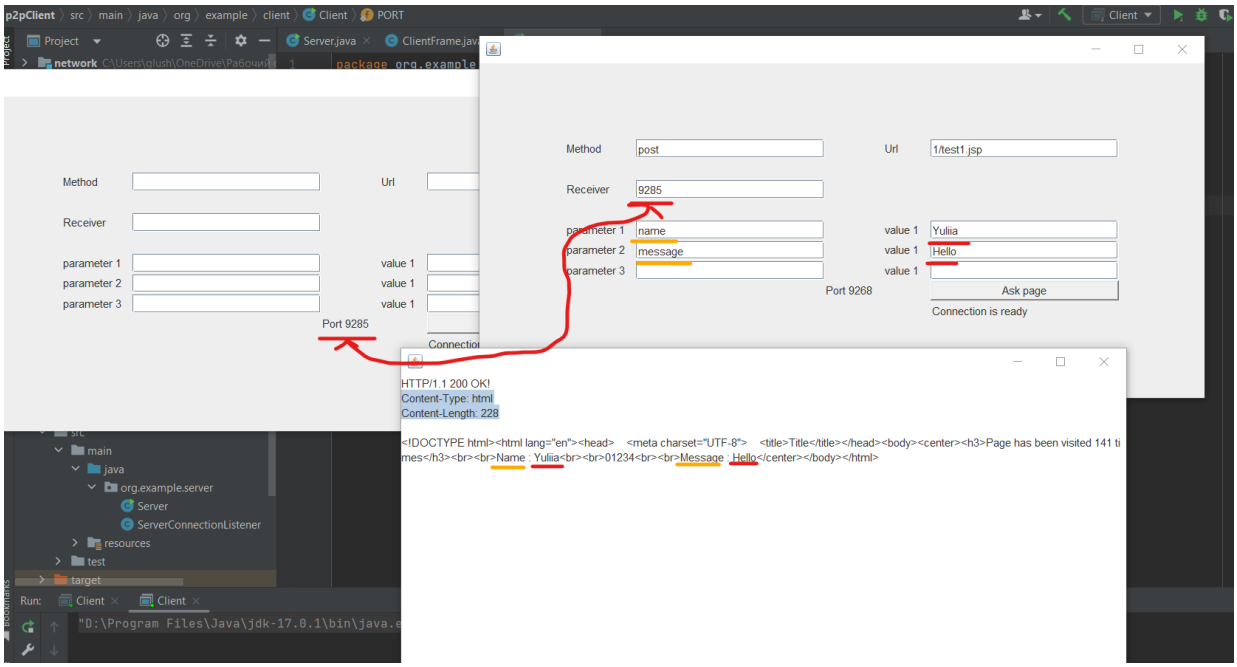


Рис.4

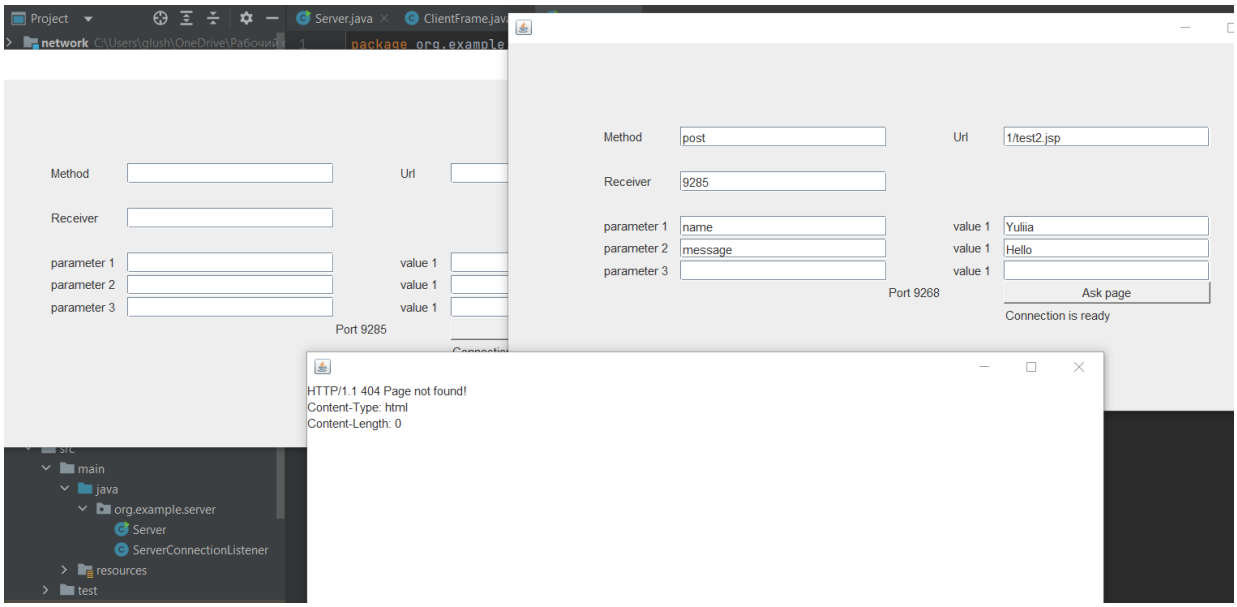


Рис.5

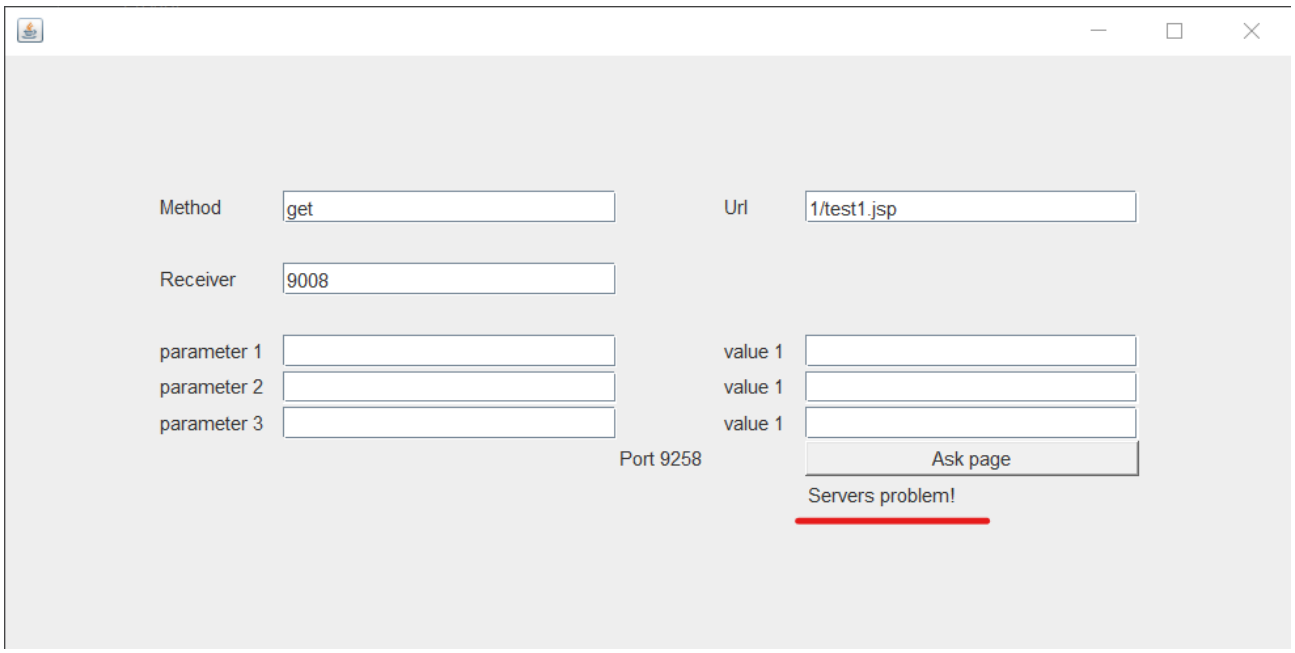


Рис.6

Сервер для пошуку активних підключень працює так: потрібно зробити запит на сервер (будь-який метод, будь-який url, замість порту вказати “server”(регістр не має значення)). Після цього дане підключення буде додано до списку активних, та надіслано йому список інших активних підключень . Якщо peer стає не активним, на сервер автоматично направляється запит, що сповіщає серверу, що дане підключення потрібно видалити із списку активних.

Рисунок 7 – підключення із Google Chrome до активного peer

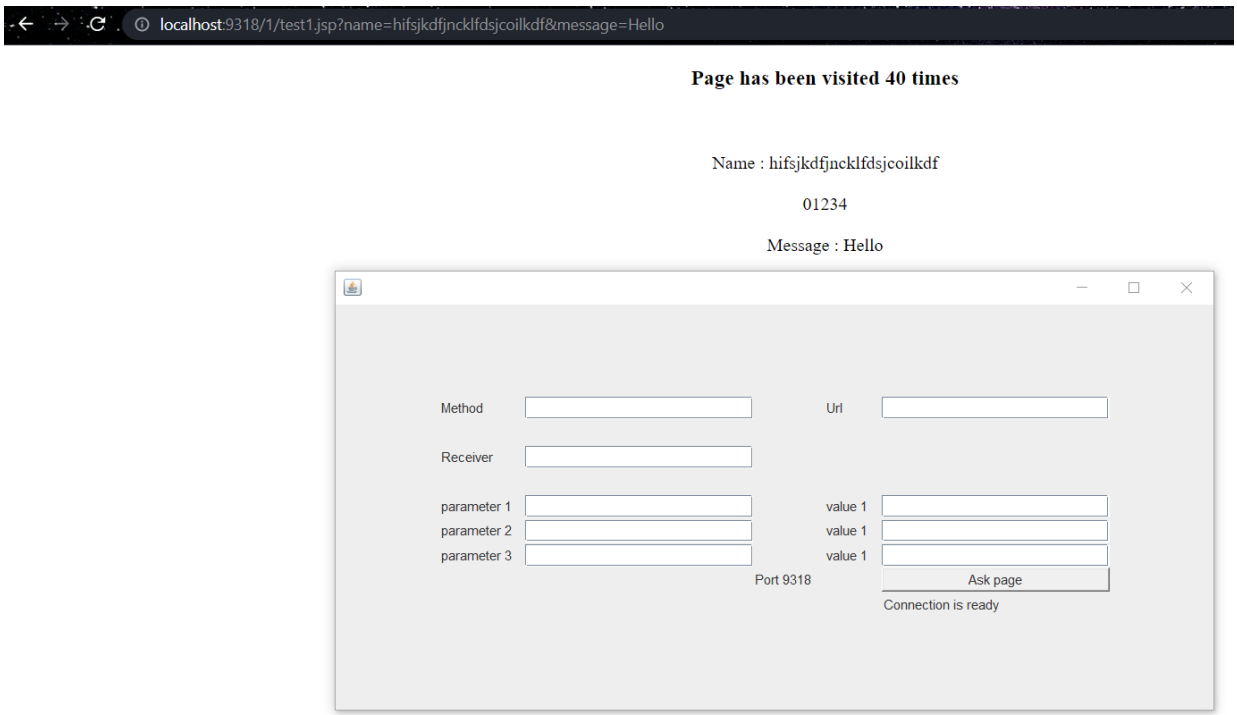


Рис.7

Висновок : під час виконання даної лабоаторної роботи було досліджено різні види взаємодії додатків, а саме client-server, peer-to-peer, service-oriented architecture. Тема лабораторного практикуму була реалізована , як peer-to-peer застосунок