



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 №2  
**Технології розроблення програмного забезпечення**  
**«ДІАГРАМА ВАРІАНТІВ ВИКОРИСТАННЯ. СЦЕНАРІЇ ВАРІАНТІВ**  
**ВИКОРИСТАННЯ. ДІАГРАМИ UML. ДІАГРАМИ КЛАСІВ.**  
**КОНЦЕПТУАЛЬНА МОДЕЛЬ СИСТЕМИ.»**

Виконала:

студентка групи ІА-23

Єрмак Д. Р

Перевірив:

Мягкий М. Ю.

## Тема лабораторних робіт:

IRC client (singleton, builder, abstract factory, template method, composite, client-server)

Клієнт для IRC-чатів з можливістю вказівки порту і адреси з'єднання, підтримка базових команд (підключення до чату, створення чату, установка імені, реєстрація, допомога і т.д.), отримання метаданих про канал.

## Завдання:

1. Ознайомитися з короткими теоретичними відомостями.
2. Проаналізуйте тему та намалюйте схему прецеденту, що відповідає обраній темі лабораторії.
3. Намалюйте діаграму класів для реалізованої частини системи.
4. Виберіть 3 прецеденти і напишіть на їх основі прецеденти.
5. Розробити основні класи і структуру системи баз даних.
6. Класи даних повинні реалізувати шаблон Репозиторію для взаємодії з базою даних.
7. Підготувати звіт про хід виконання лабораторних робіт. Звіт, що подається повинен містити: діаграму прецедентів, діаграму класів системи, вихідні коди класів системи, а також зображення структури бази даних.

## Зміст

<b>Крок 1.</b> Аналіз теми.....	2
<b>Крок 2.</b> Створення діаграми прецедентів, що відповідає обраній темі .....	3
<b>Крок 3.</b> Створення діаграми класів для частини системи .....	4
<b>Крок 4.</b> Опис 3 обраних прецедентів.....	5
<b>Крок 5.</b> Розроблення основних класів і структури системи баз даних .....	6

## Хід роботи:

### Крок 1. Аналіз теми

Для початку ми повинні проаналізувати тему та задані шаблони проектування.

IRC (Internet Relay Chat) — це протокол для комунікації в реальному часі через текстові повідомлення, що використовується для багатокористувацьких чатів (каналів) і приватних повідомлень. Клієнт для IRC дає можливість користувачам підключатися до серверів IRC для участі в чатах, використовуючи стандартні команди та функції.

Тепер розглянемо надані шаблони:

Singleton: Гарантує єдиний екземпляр клієнта IRC для керування з'єднанням з сервером, щоб уникнути створення кількох сесій одночасно.

**Builder:** Використовується для покрокового створення об'єкта з'єднання з сервером, дозволяючи налаштувати параметри.

**Abstract factory:** Дозволяє створювати родини об'єктів для різних команд IRC без вказування конкретних класів, що забезпечує інкапсуляцію окремих класів під єдиною схемою, упускаючи їхню деталізацію.

**Template method:** Визначає шаблон алгоритму для обробки основних команд IRC, де підкласи можуть змінювати окремі кроки, не змінюючи загальну структуру.

**Composite:** Структурує ієрархію об'єктів у вигляді дерева, і дозволяє уніфіковане звертання для кожного елемента дерева.

**Client-server:** Класична модель, де клієнт взаємодіє з сервером IRC через централізовану архітектуру.

**Крок 2.** Створення діаграми прецедентів, що відповідає обраній темі  
Для створення діаграми прецедентів візьмемо основні функції, описані в темі.

Варіанти використання для програми «IRC client»

1. Підключення до сервера IRC

Користувач вказує IP-адресу та порт для підключення до сервера IRC.

2. Установка імені користувача

Користувач змінює своє ім'я для участі в IRC-чатах.

3. Підключення до чату

Користувач підключається до певного каналу (чату) за допомогою команди.

4. Створення чату

Користувач створює новий канал для обговорення.

5. Отримання метаданих про канал

Користувач отримує інформацію про учасників каналу та його налаштування.

6. Реєстрація користувача

Користувач реєструється на сервері IRC, щоб отримати доступ до всіх функцій.

7. Виведення допомоги

Користувач викликає команду для отримання допомоги по основних функціях програми.

8. Відправка повідомлення

Користувач надсилає повідомлення в чат.

За визначеними варіантами використання створюємо діаграму прецедентів:

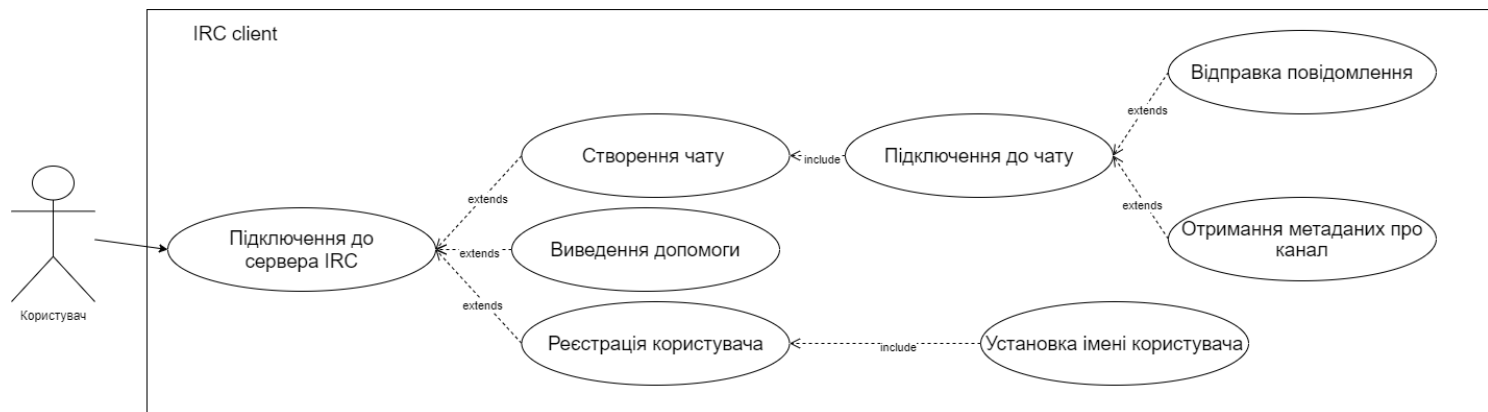


Рис. 1 – Діаграма прецедентів

### Крок 3. Створення діаграми класів для частини системи

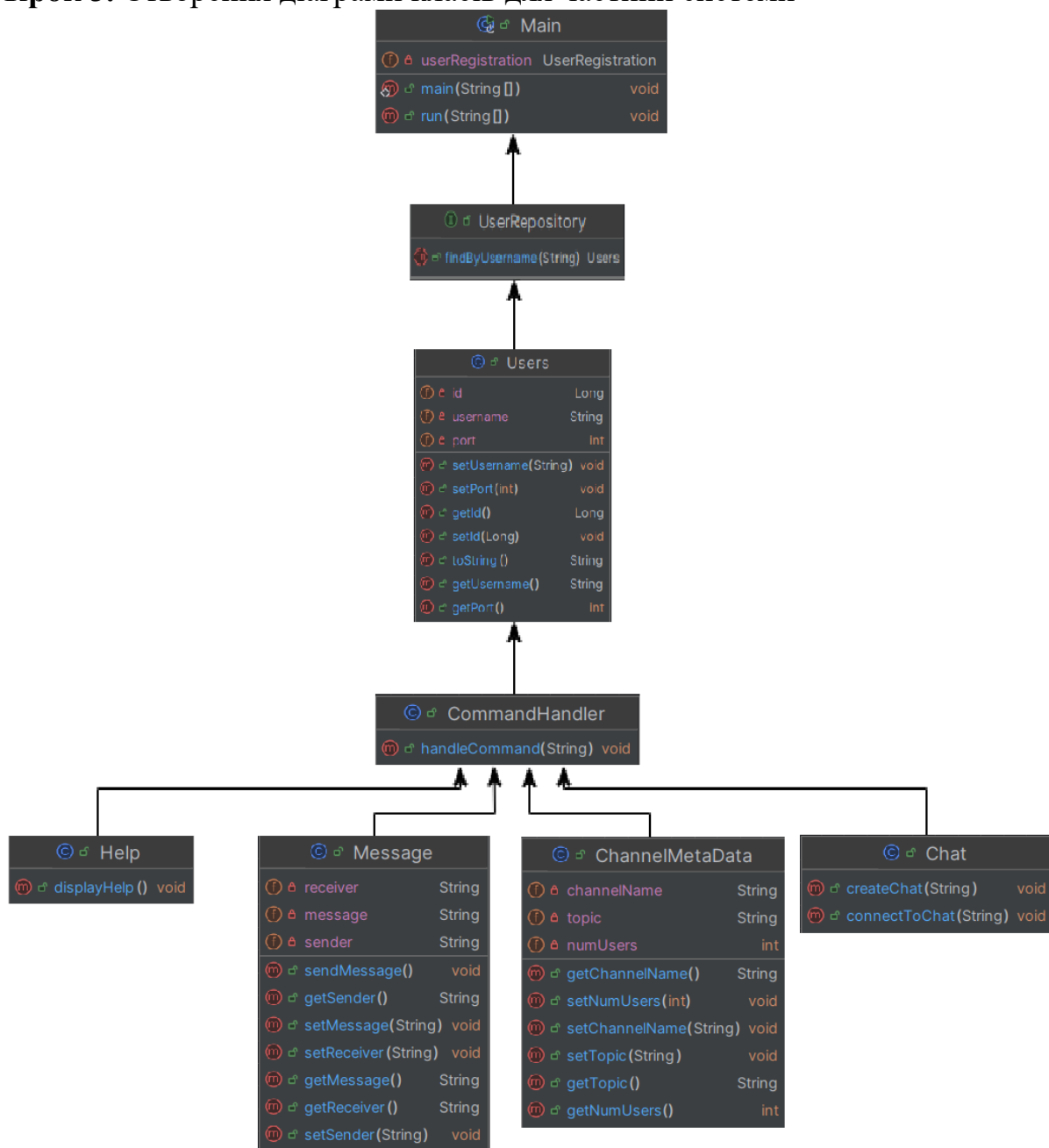


Рис. 2 – Діаграма класів

#### **Крок 4. Опис 3 обраних прецедентів**

##### **1 Прецедент:**

Назва: Реєстрація користувача

Актори: Користувач

Опис: Користувач вводить своє ім'я та порт для реєстрації. Система реєструє користувача, створює новий запис і зберігає його. Результат виводиться в консоль.

Передумови: Система запущена, користувач ще не зареєстрований.

Післяумови: Користувача зареєстровано, його дані збережено в базі даних

Виключення: Користувач ввів дані, які не є валідними. Ім'я користувача вже зайняте іншим користувачем

##### **2 Прецедент:**

Назва: Підключення до чату

Актори: Користувач

Опис: Користувач підключається до сервера, вибирає канал, і система встановлює з'єднання. Користувач під'єднується до чату і надсилає повідомлення.

Передумови: Користувач зареєстрований і підключений до сервера.

Післяумови: Користувач підключений до каналу чату, готовий до взаємодії

Виключення: Невдале підключення до сервера через неправильні параметри (IP, порт). Канал не існує. Сервер недоступний через технічні проблеми.

##### **3 Прецедент:**

Назва: Надсилання повідомлення

Актори: Користувач

Опис: Користувач створює повідомлення. Система відправляє повідомлення на сервер.

Передумови: Користувач підключений до сервера і каналу чату.

Післяумови: Повідомлення успішно надіслано і доступне для всіх учасників каналу або конкретного отримувача.

Виключення: Повідомлення не доставлено через відключення сервера або користувача. Користувач відправив порожнє повідомлення.

## Крок 5. Розроблення основних класів і структури системи баз даних

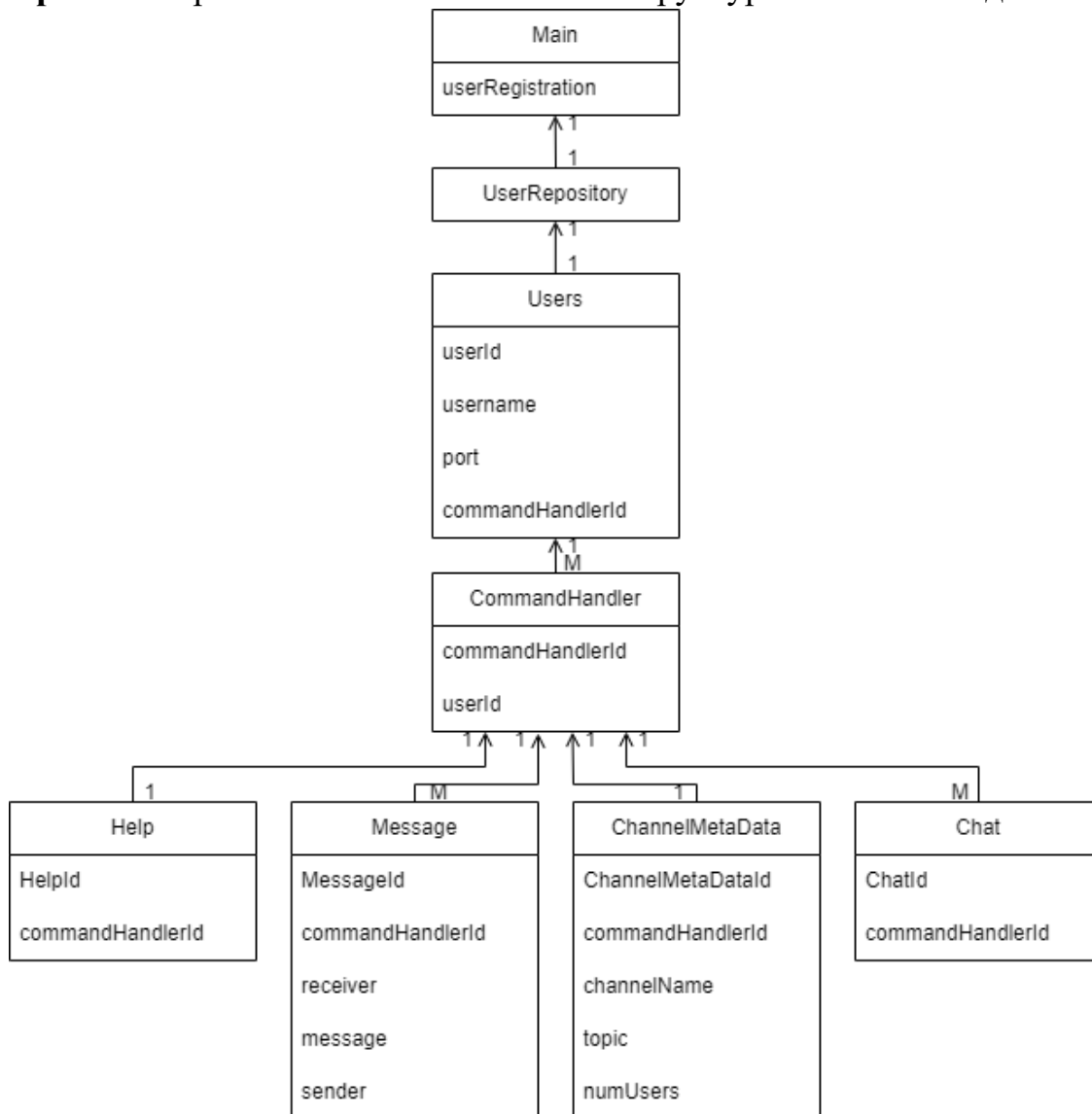


Рис. 3 – Схема таблиць бази даних

**Висновок:** У ході виконання даної лабораторної роботи було проаналізовано обрану тему роботи, шаблони проєктування та загальну структуру проєкту. Було розроблено діаграму прецедентів, основні класи майбутньої системи та схему бази даних для подальшого проєктування.