****

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  
**Технологія розроблення програмного забезпечення**

Тема: «РІЗНІ ВИДИ ВЗАЄМОДІЇ ДОДАТКІВ:

CLIENT-SERVER, PEER-TO-PEER,

SERVICE-ORIENTED ARCHITECTURE»

Варіант 25

Виконав Перевірив:

студент групи ІА-23 Мягкий Михайло

Калина С. О. Юрійович

Київ 2024

# **Тема.**

Різні види взаємодії додатків: CLIENT-SERVER, PEER-TO-PEER, SERVICE-ORIENTED ARCHITECTURE

# **Мета.**

Метою лабораторної роботи є, вивчення та практичне впровадження різних моделей взаємодії додатків, зокрема, клієнт-серверної архітектури, Peer-to-Peer (P2P) архітектури, cервіс-орієнтованої архітектури (SOA). Набуття практичних навичок реалізації розподілених систем з використанням технологій.

# **Завдання.**

1. Ознайомитися з короткими теоретичними відомостями.

2. Реалізувати частину функціоналу робочої програми у вигляді класів та їхньої взаємодії для досягнення конкретних функціональних можливостей.

3. Застосування одного з розглянутих шаблонів при реалізації програми

**Хід роботи**

**25. Installer generator (iterator, builder, factory method, bridge, interpreter, client-server)**

Генератор інсталяційних пакетів повинен мати якийсь спосіб налаштування файлів, що входять в установку, установки вікон з інтерактивними можливостями (галочка - створити ярлик на робочому столі; ввести в текстове поле деякі дані, наприклад, ліцензійний ключ і т.д.). Генератор повинен вивести один файл .exe або .msi.

# **Короткі теоретичні відомості**

#### **Клієнт-серверні додатки**

Це тип розподілених додатків, що складаються з двох основних складових:

* **Клієнт**: відповідає за інтерфейс і взаємодію з користувачем.
* **Сервер**: забезпечує зберігання, обробку даних і відповіді на запити клієнтів.

**Види клієнтів**:

1. **Тонкий клієнт**:
   * Передає більшість завдань на сервер.
   * Використовується для відображення результатів серверної обробки.
   * Підходить для задач, що вимагають централізації та безпеки.
2. **Товстий клієнт**:
   * Здатний виконувати більшу частину обробки даних на стороні клієнта.
   * Зменшує навантаження на сервер.
   * Сервер виступає лише в ролі сховища даних або точки доступу.

**Структура клієнт-серверної моделі**: Зазвичай має трирівневу архітектуру:

* **Клієнт**: відповідає за інтерфейс і дії користувача.
* **Проміжний рівень (middleware)**: включає спільні компоненти.
* **Сервер**: містить бізнес-логіку, зберігає та обробляє дані.

#### **Peer-to-Peer (P2P) додатки**

Характеризуються наступними особливостями:

* Всі учасники мережі мають рівноправний статус.
* Відсутність центрального сервера.
* Взаємодія між вузлами для досягнення спільних цілей.

**Основні виклики**:

* Забезпечення синхронізації даних між вузлами.
* Пошук інших клієнтів у мережі.

**Механізми функціонування**:

* Використання структурованих однорангових мереж.
* Протоколи обміну повідомленнями.
* Алгоритми для синхронізації даних, такі як хешування або консенсус.

#### **Сервіс-орієнтована архітектура (SOA)**

Цей підхід передбачає розробку програмного забезпечення у вигляді окремих служб:

* Компоненти системи працюють незалежно один від одного.
* Служби взаємодіють через стандартизовані інтерфейси та протоколи (наприклад, SOAP чи REST).

**Переваги**:

* Деталі реалізації компонентів приховані.
* Можливість роботи на різних платформах.
* Компоненти легко повторно використовувати.
* Архітектура добре масштабується.

#### **Мікросервісна архітектура**

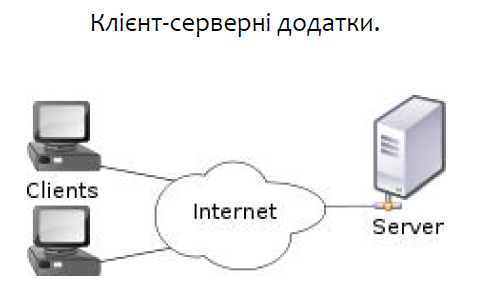
Мікросервіси — це модель побудови серверного додатку як набору незалежних компонентів:

* Кожна служба виконується у власному процесі.
* Обмін даними між сервісами здійснюється через HTTP, HTTPS, WebSockets або AMQP.
* Кожен сервіс можна оновлювати й розгортати окремо.
* Кожна служба вирішує певний набір завдань у межах своєї відповідальності.

**Переваги**:

* Гнучкість у розробці й розширенні.
* Легше масштабування великих систем.
* Полегшене обслуговування і внесення змін.

**Хід роботи**

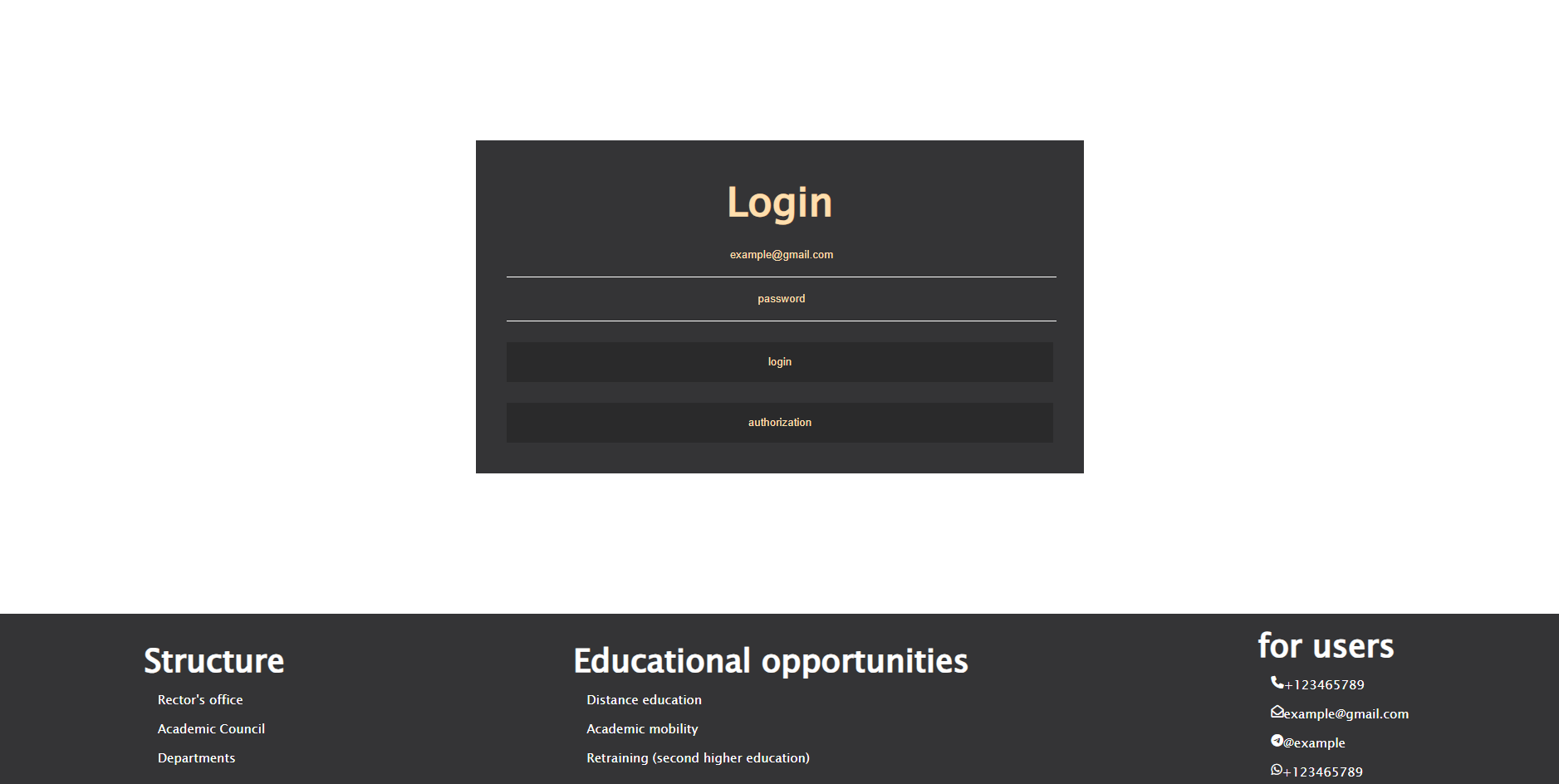


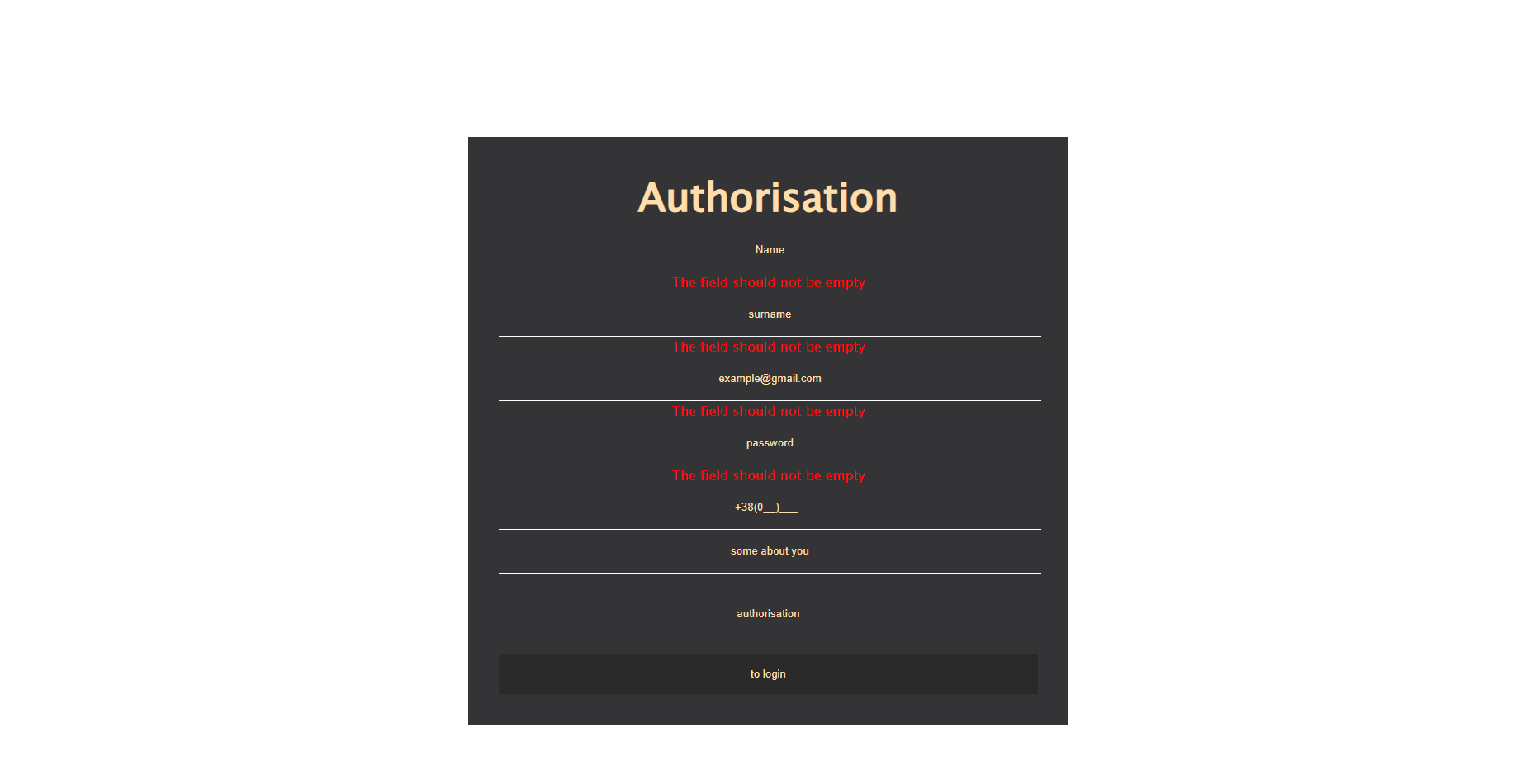
На зображенні зображено схему клієнт-серверної архітектури.

* **Клієнти** (Clients): Це комп'ютери або пристрої користувачів, які взаємодіють із сервером через Інтернет. Вони відправляють запити та отримують відповіді від сервера.
* **Інтернет** (Internet): Виступає як середовище для передачі даних між клієнтами та сервером.
* **Сервер** (Server): Центральний комп'ютер або пристрій, який обробляє запити клієнтів, зберігає дані та надсилає відповіді.

Схема ілюструє принцип взаємодії: клієнти надсилають запити до сервера через Інтернет, а сервер обробляє ці запити та повертає відповідні дані клієнтам. Це є базовою архітектурою для багатьох сучасних додатків та веб-сервісів.

**Приклад роботи**





**Висновок:**

В цій лабораторній роботі я дослідив різні види взаємодії додатків: client-server, peer-to-peer, service-oriented architecture та реалізував: client-server.