МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет “Львівська політехніка”

Інститут комп’ютерних наук та інформаційних технологій

Кафедра інформаційних систем та мереж



**Звіт**

до лабораторної роботи № 2

**«Розроблення програмних засобів для інтеграції даних із використанням SOAP повідомлень»**

з дисципліни

**«Технології інтеграції інформаційних ресурсів»**

Виконав: ст. гр. ІТІС-11

Стельмах В. Д.

Прийняв:

Щербак С.С.

Львів – 2025

**Лабораторна робота № 2**

**Тема:** Розроблення програмних засобів для інтеграції даних із використанням SOAP повідомлень

**Мета роботи:** Ознайомитись із особливостями створення та дослідження SOAP повідомлень для роботи із вебсервісами шляхом розробки власної програми-клієнта.

**Порядок виконання роботи:**

1. Налаштувати середовище розробки Python та встановити необхідні бібліотеки (зокрема zeep).
2. Розробити програму на Python, яка взаємодіє з вебсервісом TemperatureConversions (https://webservices.daehosting.com/services/TemperatureConversions.wso?wsdl).
3. Реалізувати функції для виклику кожної операції вебсервісу (CelsiusToFahrenheit, FahrenheitToCelsius, WindChillInCelsius, WindChillInFahrenheit).
4. Реалізувати надсилання SOAP-запитів та отримання/парсинг SOAP-відповідей за допомогою бібліотеки zeep.
5. Реалізувати обробку можливих помилок, включаючи SOAP Faults та помилки валідації/транспорту, з використанням механізму логування.
6. Продемонструвати роботу програми шляхом виклику операцій з різними параметрами, включаючи некоректні дані для тестування обробки помилок.
7. Задокументувати результати роботи програми (консольний вивід) та надати посилання на репозиторій з кодом програми у звіті.

**Короткі теоретичні відомості**

**SOAP (Simple Object Access Protocol):** Протокол обміну структурованими повідомленнями на основі XML, що використовується для взаємодії вебсервісів. Стандартне SOAP-повідомлення має структуру "конверта" (Envelope), який містить необов'язковий заголовок (Header) та обов'язкове тіло (Body). У тілі передаються дані запиту або відповіді. У разі помилки на сервері, тіло містить елемент Fault, що описує помилку.

**Взаємодія з SOAP-сервісом програмно:** Для взаємодії з SOAP-сервісом з програмного коду необхідно:

1. Мати WSDL-опис сервісу для розуміння його структури, операцій та типів даних.
2. Сформувати коректне SOAP-повідомлення (запит) у форматі XML.
3. Надіслати цей запит на URL-адресу сервісу за допомогою HTTP POST запиту з відповідними заголовками (напр., Content-Type: text/xml або application/soap+xml).
4. Отримати HTTP-відповідь від сервера.
5. Розпарсити тіло відповіді (XML), щоб витягти результат операції або інформацію про помилку (SOAP Fault).

**Бібліотека zeep для Python:** Сучасна та швидка бібліотека Python для роботи з SOAP-клієнтами. Вона значно спрощує процес взаємодії:

* Автоматично завантажує та аналізує WSDL-файл.
* Надає об'єктний інтерфейс для виклику операцій сервісу як звичайних методів Python.
* Автоматично генерує SOAP-запити на основі переданих параметрів.
* Автоматично парсить SOAP-відповіді та повертає результати у вигляді Python-об'єктів.
* Обробляє помилки SOAP Fault, перетворюючи їх на винятки Python (zeep.exceptions.Fault).
* Виконує валідацію даних на стороні клієнта відповідно до типів, визначених у WSDL.

**Логування (Logging):** Процес запису інформації про події, що відбуваються під час роботи програми. Логи допомагають у відстеженні виконання програми, діагностиці помилок та моніторингу. У Python для цього використовується стандартний модуль logging.

**Результати виконання роботи**

**Протокол роботи програми**

Було розроблено та запущено Python-скрипт, який послідовно викликав операції вебсервісу TemperatureConversions з різними тестовими даними, а також симулював помилкові ситуації. Нижче наведено повний консольний вивід програми (логи) під час її виконання:

Зображення, що містить текст, знімок екрана, програмне забезпечення

Вміст, створений ШІ, може бути неправильним.

**Аналіз протоколу роботи:**

* Програма успішно створила SOAP-клієнт для WSDL-адреси <https://webservices.daehosting.com/services/TemperatureConversions.wso?wsdl>
* Успішно виконано виклики всіх чотирьох операцій (CelsiusToFahrenheit, FahrenheitToCelsius, WindChillInCelsius, WindChillInFahrenheit) з коректними параметрами. Логи показують параметри виклику та успішне завершення. Результати, виведені на консоль (частина print), відповідають очікуваним значенням конвертації та розрахунків.
* Продемонстровано обробку помилки валідації даних при спробі передати нечислове значення (наприклад, "invalid\_temperature") в операцію convert\_celsius\_to\_fahrenheit. Логер зафіксував помилку ValueError та повідомив про невалідний вхідний параметр.
* Логування дозволило відстежити кожен крок взаємодії з вебсервісом, включаючи спроби виклику, передані параметри та інформацію про успішне виконання або виникнення помилок.

**Посилання на вихідний код програми**

Повний вихідний код розробленої програми на Python розміщено у репозиторії GitHub за наступним посиланням:

<https://github.com/VitaliiStelmakh/TIIR/blob/main/lab_2/lab_2.py>

**Висновки**

Під час виконання лабораторної роботи я ознайомився(лась) з особливостями програмної взаємодії з SOAP вебсервісами та розробив(ла) клієнтський застосунок на Python. Було застосовано бібліотеку zeep для спрощення роботи з WSDL, формування запитів та обробки відповідей, а також модуль logging для відстеження виконання та діагностики помилок. Здобуто практичні навички інтеграції даних шляхом виклику операцій вебсервісу та обробки його результатів і можливих помилок.