

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**Український державний університет науки і технологій**

Кафедра «Комп’ютерні інформаційні технології»

**Лабораторна робота №6**

**з дисципліни «Компонентні технології програмного забезпечення»**

**на тему:**

**«**Розробка web-сервісів.**»**

Виконала:

Студент гр. ПЗ2011

Кулик С.В.

Прийняв:

Андрющенко В. О.

Дніпро, 2023

**Тема.** Розробка web-сервісів.

**Теоретичні відомості**

***Основні положення моделі веб-сервісів***

Веб-сервіси є концепцією створення таких додатків, функції яких можна використовувати за допомогою стандартних протоколів Інтернет. В даний час цю концепцію застосовують і розвивають багато провідних компаній в ITгалузі. Концепція веб-сервісів реалізується за допомогою ряду технологій, які стандартизовані World Wide Web Consortium (W3C).

Веб-сервіси є одним з варіантів реалізації компонентної архітектури. XML є фундаментом для створення більшості технологій, пов'язаних з вебсервісами. Для віддаленої взаємодії з веб-сервісами використовується Simple Object Access Protocol (SOAP). SOAP забезпечує взаємодію розподілених систем, незалежно від об'єктної моделі, операційної системи або мови програмування. Дані передаються у вигляді особливих XML документів особливого формату.

Згідно з визначенням W3C, веб-сервіси це додатки, які доступні по протоколах, які є стандартними для Інтернет. Немає вимоги, щоб веб-сервіси використовували якийсь певний транспортний протокол. Специфікація SOAP визначає, яким чином пов'язуються повідомлення SOAP і транспортний протокол.

Найбільш часто реалізується передача SOAP повідомлень по протоколу HTTP. Також широко поширене використання в якості транспортного протоколу SMTP, FTP, TCP.

Згідно з визначенням W3C, WSDL - формат XML для опису мережевих сервісів як набору кінцевих операцій, які працюють за допомогою повідомлень, що містять документно-орієнтовану або процедурно-орієнтовану інформацію. Документ WSDL повністю описує інтерфейс веб-сервісу із зовнішнім світом. Він надає інформацію про послуги, які можна отримати, скориставшись методами сервісу, і способах звернення до цих методів.

Технологія Universal Description, Discovery and Integration (UDDI) передбачає ведення реєстру веб-сервісів. Підключившись до цього реєстру, споживач зможе знайти веб-сервіси, які найкращим чином задовольняють його потребам. Технологія UDDI дає можливість пошуку і публікації потрібного сервісу, як людиною, так і програмою-клієнтом. Пошук і публікація в реєстрі надається програмі-клієнту як набір веб-сервісів реєстру UDDI.

Веб-сервіси позиціонуються як програмне забезпечення проміжного шару. Використовувати веб-сервіси можуть як клієнтські програми, які безпосередньо працюють з користувачем, так і інші додатки (в тому числі і інші веб-сервіси). Веб-сервіси розміщуються на серверах додатків.

Розробники концепції веб-сервісів пропонують наступні сценарії застосування веб-сервісів:

* Веб-сервіси як реалізація логіки додатка (бізнес-логіки). Тобто, створення нового додатка, бізнес-логіка якого реалізується в веб-сервісі.
* Веб-сервіси як засіб інтеграції. Тобто, використання веб-сервісу як способу доступу віддалених клієнтів до внутрішньої ІС компанії, або для організації взаємодії компонента (наприклад, EJB, COM-компонента) з різними віддаленими клієнтами.
* Створення альтернативного сервісу. В цьому випадку при розробці нового вебсервісу використовується опис інтерфейсу вже існуючого веб-сервісу. Таким чином, сервіс має багато потенційних клієнтів відразу з моменту початку роботи, а підключення до нього не вимагає істотних змін на стороні клієнта.

Як було сказано вище, концепція веб-сервісів включає в себе можливість ведення реєстру веб-сервісів. Опис інтерфейсу може бути отримано з такого реєстру. Після створення і впровадження нового веб-сервісу має сенс зареєструвати його в реєстрі. Тоді клієнти при пошуку сервісів, що реалізують вихідний інтерфейс, отримають вказівку і на новий веб-сервіс.

Використання веб-сервісу як будівельного блоку при створенні програми. Додаток може використовувати веб-сервіси як віддалені компоненти, які надають певну функціональність. Існують різні сервіси, які надають якісне рішення таких завдань як аутентифікація, ведення календаря, відправлення повідомлень, пошук, переклад тощо.

Крім цього, можливі інші варіанти використання веб-сервісів. Наприклад, існують дослідження з використання веб-сервісів для побудови розподілених обчислювальних і інформаційних систем і тимчасових, і зі складною ієрархічною структурою.

**Текст програми**

**Файл HelloWebService.java**

package com.example.ktpz6;  
  
import javax.jws.WebMethod;  
import javax.jws.WebService;  
import javax.jws.soap.SOAPBinding;  
  
@WebService  
@SOAPBinding(style = SOAPBinding.Style.*RPC*)  
public interface HelloWebService {  
 @WebMethod  
 public String getHelloString(String name);  
}

**Файл HelloWebServiceImpl.java**

package com.example.ktpz6;  
  
import javax.jws.WebService;  
  
@WebService(endpointInterface = "com.example.ktpz6.HelloWebService")  
public class HelloWebServiceImpl implements HelloWebService {  
 @Override  
 public String getHelloString(String name) {  
 return "Hello, " + name + "!";  
 }  
}

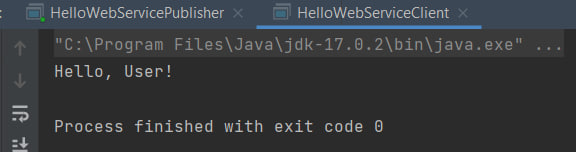
**Файл HelloWebServicePublisher.java**

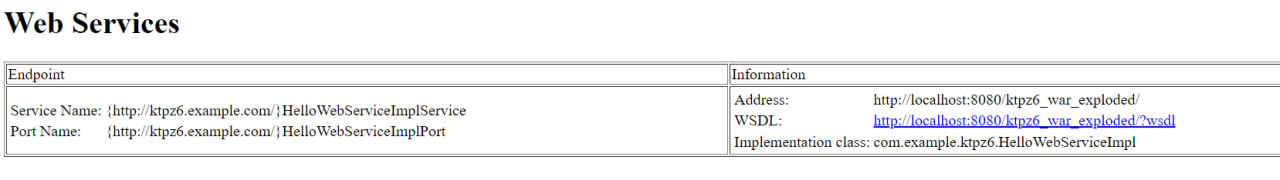
package com.example.ktpz6;  
  
import javax.xml.ws.Endpoint;  
  
public class HelloWebServicePublisher {  
 public static void main(String... args) {  
 Endpoint.*publish*("http://localhost:8080/ktpz6\_war\_exploded/", new HelloWebServiceImpl());  
 }  
}

**Файл HelloWebServiceClient.java**

package com.example.ktpz6;  
  
import java.net.URL;  
import java.net.MalformedURLException;  
import javax.xml.namespace.QName;  
import javax.xml.ws.Service;  
  
public class HelloWebServiceClient {  
 public static void main(String[] args) throws MalformedURLException {  
 URL url = new URL("http://localhost:8080/ktpz6\_war\_exploded/");  
 QName qname = new QName("http://ktpz6.example.com/","HelloWebServiceImplService");  
 Service service = Service.*create*(url, qname);  
 HelloWebService hello = service.getPort(HelloWebService.class);  
 System.*out*.println(hello.getHelloString("User"));  
 }  
}

**Результати роботи програми**





**Висновок**

Розробка web-сервісів є критично важливим етапом у сучасному програмуванні, оскільки все більше людей користуються Інтернетом і залежать від нього у повсякденному житті. Web-сервіси виступають ключовими складовими для забезпечення ефективної та зручної роботи Інтернету.

Для успішної розробки web-сервісів необхідні технічні знання і навички, такі як вміння програмувати, вміння проектувати архітектуру додатків та розуміння протоколів обміну даними, таких як HTTP та REST.

Основною метою розробки web-сервісів є створення зручного та простого інтерфейсу, який забезпечує надійний обмін даними між клієнтами та серверами. Для цього необхідно розуміти потреби користувачів та можливості серверів.

Розробка web-сервісів може мати різноманітні застосування, такі як створення електронного магазину, веб-додатка, веб-сервісу для обміну даними між різними системами та інших. При цьому важливо забезпечити безпеку та надійність обміну даними, а також оптимізувати продуктивність додатків.