

TP N° 2 : Services Web

Services Web étendus (WS-*)

I. Mouakher

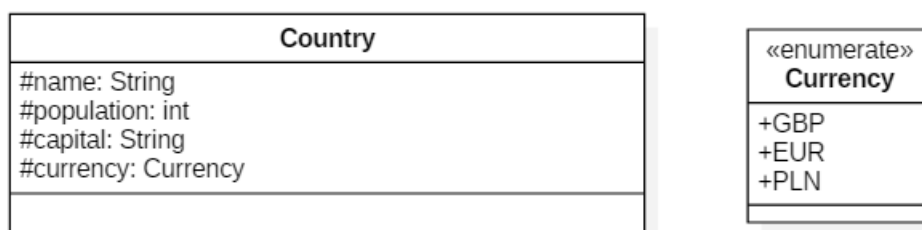
Classe : CII-2-GLSI-N

Buts pédagogiques : Création et consommation de web services SOAP en utilisant Spring Boot.

Description :

Produire un web service

Dans cette section, nous allons utiliser l'approche Bottom-Up pour créer un web service qui propose une simple opération `getCountry`. Le paramètre input de cette opération est `getCountryRequest` de type string qui contient le nom du Pays. Le paramètre output de cette opération est `getCountryResponse` de type complexe : `country`. L'opération `getCountry` retourne toutes les informations sur un pays donné.



- Créer un Application Spring Boot **producing-web-service** pour fournir ce service Web
 - JDK 1.8, IntelliJ, Maven – Development environment
 - Spring-boot – Underlying application framework
 - [wsdl4j](#) – for publishing WSDL for our Service
 - Postman – for testing our service
 - [JAXB maven plugin](#) – for code generation
- Créer le fichier XSD dans le répertoire ressource de votre projet, pour générer ensuite le WSDL `countries.wsdl`.
- Implémenter la classe `CountryRepository` (`@Component`) dans laquelle on va utiliser un objet `countries` : `private static final Map<String, Country> countries = new HashMap<>();`
et deux operations :
 - `public void initData()` pour initialier l'objet `countries`. (`@PostConstruct`)
 - `public Country findCountry(String name)` pour retourner les différentes information sur un pays donné.
- Créer l'endpoint du service en développant la classe `CountryEndpoint` avec comme URI : <http://www.tekup.de/country/ws/ProducingWebService>
- Configurer les Beans du Web Service en développant la classe `WebServiceConfig`. Le service est disponible à l'URL : <http://localhost:8080/ws/>. Le `TargetNamespace` est <http://www.tekup.de/country/ws/ProducingWebService>. Le `PortTypeName` est "CountriesPort".

6. Tester Invocation du web service.

Vous pouvez maintenant exécuter votre application, une fois le serveur lancé, vous pouvez tester le web service et consulter le WSDL dans un navigateur. Utiliser Postman pour invoquer l'opération getCountry de votre service.

Consommer un web service

Dans cette section, nous voyons comment consommer un web service à partir d'une application Spring Boot. On peut considérer un consommateur de web service comme une application cliente (au sens de l'architecture Client-Serveur), ou bien comme un Mashup, c'est-à-dire une application utilisant des services provenant d'une ou plusieurs sources pour proposer un contenu (ou un service).

1. Créer une Application Spring Boot **consuming-web-service**
2. Test du client Après s'être assuré que le web service est toujours lancé. Construire le fichier jar et ensuite exécuter le : `java -jar target/consuming-web-service-0.0.1-SNAPSHOT.jar` Poland