

Objectifs :

- (1)- Modélisation d'une architecture SOA (2)- Application des principes du style SOA

Activité 1

Le processus de « review » (examen des papiers scientifiques) dans une conférence ou congrès scientifique se déroule comme suit : (1) Le comité de programme annonce un "appel à communications". (2) Les auteurs reçoivent l'appel à communications et décident de présenter des papiers décrivant leurs travaux scientifiques. Ils les envoient au comité d'organisation de la conférence. Un papier donné peut avoir plusieurs auteurs, mais un seul auteur de correspondance. (3) Le président du comité de programme enregistre les papiers reçus en leur attribuant un numéro d'anonymat. (4) Il affecte les papiers aux membres du comité de scientifique pour la relecture ou arbitrage (on parle de processus d'arbitrage, review, ou d'examen). Chaque papier (anonyme) est envoyé à trois arbitres (examineurs) distincts. (5) Le président comité de programme recueille les rapports des arbitres (reviewers). (6) À base de ces rapports, le comité de programme se réunit et sélectionne les papiers acceptés pour l'inclusion dans le programme et notifie les auteurs à propos de la sélection. (7) Certains papiers peuvent être sujet à des modifications par les auteurs qui doivent les resoumettre pour avis supplémentaires des arbitres qui ont exigé ces modifications. (8) Le président du comité de programme notifie aux auteurs des résultats de la sélection.

Les auteurs (principalement l'auteur de correspondance) doit créer un compte pour pouvoir soumettre l'article et suivre les différentes étapes de son papier.

L'organisme (un centre de recherche) qui organise ce séminaire biannuel, possède une collection d'applications :

- Une BDD sous ORACL, qui permet de gérer les rapports et papiers scientifiques produit par l'organisme. Cette BD possède un application cliente permettant : Les opérations de bases sur les rapports et leurs modèles (ajout, suppression, modification, versioning) ; la classification, la recherche ; la notification, l'annotation et Les statistiques
- Un serveur courriel (mail) qui permet d'héberger les mails de l'organisme. Ce serveur possède aussi une application cliente permettant la : Gestion les mail (lecture, création, envoi réception ; Recherche et classification et archivage des mails et des pièces jointes dans la BD ; Envoi de notification par sms aux utilisateurs inscrit dans la bd
- Une application portail web qui offre la possibilité aux chercheurs et visiteur de l'organisme d'avoir accès via le web à : Les dernières infos et événements de l'organisme ; Les informations utiles par activité de l'organisme ; Soumettre en ligne des rapports des recherches effectué par les chercheurs affiliés à l'organisme enregistrés dans la BD.

On désire avoir un système de gestion de conférences pour notre organisme, il vous est demandé d'appliquer l'approche par processus suivante :

1. Concevoir l'architecture SOA de ce système en suivant l'approche par processus (utiliser le diagramme d'activité d'UML pour modéliser les processus)
2. Donner la vue composant/connecteur de l'architecture de SOA du système
3. Donner la vue déploiement de ce système

Activité 2 .

1. Etudier la description WSDL suivante :
 - a. Créer le/les services en utilisant le style webservice soap (en utilisant .NET et JEE)
 - b. Donner l'équivalent REST de ce document et créer le ou les services REST décrit en utilisant une technologie telle que NODE.JS
2. Donner la vue C&C d'un système qui utilise ces web services

```
<definitions name="shopCart" targetNamespace=http://www.cart.com/wsd/ShopCart.wsd/ xmlns:tns="http://www.cart.com/wsd/ShopCart.wsd/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wsdl:message name="contentReq" />
  <wsdl:message name="contentSupply">
    <wsdl:part name="contentSupply" type="apachesoap:Vector" />
  </wsdl:message>
  <wsdl:message name="iteminReq">
    <wsdl:part name="itmid" type="xsd:int" />
    <wsdl:part name="itmnr" type="xsd:double" />
  </wsdl:message>
  <wsdl:message name="itemoutReq">
    <wsdl:part name="itmid" type="xsd:int" />
  </wsdl:message>
  <wsdl:portType name="cart">
    <wsdl:operation name="cartlist">
      <wsdl:input message="impl:contentReq" name="contentReq" />
      <wsdl:output message="impl:contentSupply" name="contentSupply" />
    </wsdl:operation>
    <wsdl:operation name="itemin" parameterOrder="itmid","itmnr">
```

```
<wsdl:input message="impl:iteminReq" name="iteminReq" />
<wsdl:output message="impl:iteminRsp" name="iteminRsp" />
</wsdl:operation>
<wsdl:operation name="itemout" parameterOrder="itmId">
  <wsdl:input message="impl:itemoutReq" name="itemoutReq" />
  <wsdl:output message="impl:itemoutRsp" name="itemoutRsp" />
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="ShopCartSoapBinding" type="impl:Compte">
  <wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="cartlist">
    <wsdlsoap:operation soapAction="contentReq" />
    <wsdl:input name="contentReq">
      <wsdlsoap:body
        encodingStyle=http://schemas.xmlsoap.org/soap/encoding/ namespace=http://localhost:8080/axis/services/ShopCart use="encoded" />
    </wsdl:input>
    <wsdl:output name="contentSupply">
      <wsdlsoap:body encodingStyle=http://schemas.xmlsoap.org/soap/encoding/ namespace=http://localhost:8080/axis/services/ShopCart use="encoded" />
    </wsdl:output>
  </wsdl:operation>
<wsdl:operation name="itemin">
  <wsdlsoap:operation soapAction="itemin" />
  <wsdl:input name="iteminReq">
    <wsdlsoap:body encodingStyle=http://schemas.xmlsoap.org/soap/encoding/ namespace="http://localhost:8080/axis/services/ShopCart" use="encoded" />
  </wsdl:input>
  <wsdl:output name="iteminRsp">
    <wsdlsoap:body encodingStyle=http://schemas.xmlsoap.org/soap/encoding/ namespace=http://localhost:8080/axis/services/ShopCart use="encoded" />
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="itemout">
  <wsdlsoap:operation soapAction="itemout" />
  <wsdl:input name="itemoutReq">
    <wsdlsoap:body encodingStyle=http://schemas.xmlsoap.org/soap/encoding/ namespace="http://localhost:8080/axis/services/ShopCart" use="encoded" />
  </wsdl:input>
  <wsdl:output name="itemoutRsp">
    <wsdlsoap:body encodingStyle=http://schemas.xmlsoap.org/soap/encoding/ namespace=http://localhost:8080/axis/services/ShopCart use="encoded" />
  </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="shopCartService">
  <wsdl:port binding="impl:ShopCartSoapBinding" name="shopCartService">
    <wsdlsoap:address location="http://localhost:8080/axis/services/shopCartsBb" />
  </wsdl:port> </wsdl:service> </definition>
```